ALIBABA CLOUD

Alibaba Cloud

Elastic Compute Service

Tutorials

Document Version: 20220711

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.Quick start

This topic describes how to build websites on Elastic Compute Service (ECS) instances.

Procedure

1. Select an ECS instance.

   Different types of websites have different requirements on instance configurations. Select an appropriate instance based on the website size and the number of potential visitors. In most cases, you can select instances that have basic configurations for small websites. We recommend that you select an instance in the same way as you purchase an instance. For more information, see Create an instance by using the wizard.

   The following table describes recommended instance types and disks.

Configuration	Recommendation
Instance type	◦ Instance types in the s6 shared standard and t6 burstable instance families. These instance types are cost-effective and meet basic requirements on website building. ◦ Instance types in the c6 compute-optimized and g6 general-purpose instance families. These instance types deliver high performance and are suitable for website building scenarios that require reliable instance performance. For more information about how to select instance families and instance types, see Instance family and Best practices for instance type selection. You can upgrade or downgrade an instance based on your needs. For more information, see Instance families that support instance type changes.
Disk	◦ Performance level 0 (PL0) enhanced SSDs (ESSDs): use the next-generation distributed Elastic Block Storage (EBS) architecture to deliver ultra-high performance. For more information, see ESSDs. ◦ Ultra disks: provide cost-effectiveness, medium random IOPS, and high reliability. For more information about the performance metrics and specifications of disks, see EBS performance.

2. Configure rules for the security group.

   By default, ports 22 and 3389 that are required to connect to an instance are enabled when you create a security group. For this step, make sure that these ports are enabled to allow inbound traffic. If these ports are not enabled, manually configure security group rules to allow inbound traffic on these ports. For more information, see Add a security group rule.

3. Deploy a website.

Select a method to deploy your website. For more information, see Summary of website building methods.

4. Purchase a domain name.

   Enter a domain name that you want to purchase. If the domain name is not in use, you can purchase it. For more information, see Register a generic domain nameHow to register an Alibaba Cloud domain name.

   For information about the differences between the .com and .net suffixes, see Domain name differences.

5. Apply for an Internet Content Provider (ICP) filing for the domain name.

   > **Note** If the instance that hosts your website is located in a region within the Chinese mainland, you must apply for an ICP filing for your domain name. Otherwise, you can skip this step.

   i. Prepare for the ICP filing.

      For more information about ICP filing regulations, see ICP filing regulations of MIIT in different regions. For more information about preparations for ICP filings, see Overview.

   ii. Apply for the ICP filing.

6. Resolve the domain name.

   You can resolve your domain name in Alibaba Cloud DNS. For more information, see Configure domain name resolution. After you configure the domain name resolution, users can visit your website by using the configured domain name.

   To map the domain name to an IP address, add an A record. For more information, see Record types.

7. (Optional) Enable secure HTTPS access.

   SSL Certificates Service allows you to redirect traffic to your websites or mobile applications from HTTP to HTTPS at minimal costs. You can use SSL certificates to authenticate users and encrypt data. For more information, see What is Certificate Management Service? If you purchase and download an SSL certificate, the methods of installing the certificate on severs in different environments vary. For more information, see Installation overview.

After you perform the preceding steps, a website is built on your own. You can use the domain name to visit the website and check whether the service is normal.

Billing

An ECS instance includes computing resources (vCPUs and memory), an image, EBS devices, public bandwidth, and snapshots. You are charged for these resources.

The following billing methods are supported:

- Subscription: You pay for resources upfront and use them over a specified period of time.
- Pay-as-you-go: You pay for resources after you use them. Resources can be purchased and released as needed.
- Preemptible instance: You can bid for available computing resources to create preemptible instances. Preemptible instances offer discounts compared with pay-as-you-go instances. However, preemptible instances can be reclaimed.

- Reserved instance: Reserved instances are discount coupons that are used together with pay-as-you-go instances. When you purchase a reserved instance, you make a commitment to use instances that have specified configurations such as instance type, region, and zone to receive discounted billing. Reserved instances are applied to offset the bills of computing resources.
- Savings plan: Savings plans are discount plans that are used together with pay-as-you-go instances. When you purchase a savings plan, you make a commitment to use a consistent amount (measured in USD/hour) of resources to receive discounted billing. Saving plans are applied to offset the bills of computing resources and system disks.
- Storage capacity unit (SCU): SCUs are storage resource plans provided for use with pay-as-you-go storage resources. When you purchase an SCU, you make a commitment to use storage resources of specific capacity to receive discounted billing. SCUs are applied to offset the bills of various storage resources such as EBS devices, Apsara File Storage NAS file systems, and Object Storage Service (OSS) buckets.

For more information about the billing methods of ECS instances, see Overview and the Pricing tab of the Elastic Compute Service product page.

FAQ

The following section provides answers to frequently asked questions about using ECS instances or building websites:

Security groups and snapshots

- Security groups for different use casesConfiguration guide for ECS security groups
- Roll back a disk by using a snapshot

Website access failures

- What are the common causes of and solutions to failures that occur when I attempt to visit my website?
- How do I test the connectivity when the ping result shows packet loss or when the ping operation fails?

References

- For information about how to select Alibaba Cloud services and configurations based on your business needs, see Architecture Consulting Service.
- If you want to migrate your business from your data center or a hosted data center to Alibaba Cloud, you can request technical support from Alibaba Cloud for cloud migration. For more information, see Cloud Migration Service.

2.Summary of website building methods

This topic describes the methods used to deploy different types of websites.

Website type	Deployment method	Description
WordPress	• Create a WordPress environment by using ROS • Manually build a WordPress website on a CentOS 7 ECS instanceManually build a WordPress website on a CentOS 7 ECS instance • Manually build a WordPress website on a Windows ECS instance	WordPress is a common software program used to build personal blogs, websites, and apps. WordPress is a content management system (CMS) that you can use to build and maintain your websites. Images or Resource Orchestration Service (ROS) templates can be used to deploy WordPress. This solves space and programming problems and simplifies website building processes. ROS uses JSON-formatted template files to create Alibaba Cloud stacks. You can also build WordPress manually.
LNMP environment	• Use ROS to deploy an LNMP environment • Manually build an LNMP environment in CentOS 6 • Manually build an LNMP environment on a CentOS 7 instance	LNMP is an acronym of the names of its original four open source components: the Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. Images or ROS templates can be used to deploy LNMP environments. If you are familiar with the Linux operating system, you can deploy LNMP environments manually to meet your requirements.

Website type	Deployment method	Description
Java web environment	• Manually deploy a Java web environment on a CentOS 7 instance • Use Cloud Toolkit to deploy a Java web environment	Tomcat is an open source Java web server that is used as a web development tool. Tomcat can host Java Web applications that consist of Servlet, JavaServer Pages (dynamic content), HTML pages, JavaScript, Stylesheet, and images (static content). • Manually deploy a Java web environment. If you are familiar with Linux commands, you can manually deploy Java web projects on ECS instances to meet your requirements. • Deploy a Java web environment by using a plug-in. Alibaba Cloud Toolkit for Eclipse (Cloud Toolkit) is a free plug-in used for integrated development environment (IDE). After you develop, debug, and test an application on the premises, you can use this plug-in to deploy the application to an ECS instance.
Node.js	• Deploy a Node.js environment on a CentOS 7 instance	Node.js is a JavaScript runtime built on the Chrome V8 JavaScript engine. You can use Node.js to build online applications and implement extensions. Node.js uses an event-driven and non-blocking I/O model. This lightweight and efficient model is suitable for data-intensive real-time applications that run on distributed devices. The Node.js package manager (npm) is the largest ecosystem of open source libraries in the world.
Magento e-commerce website	Build a Magento e-commerce website on ECS	Magento is an open source e-commerce solution that has a modular architecture and varied expansion features.
Microsoft SharePoint 2016	Build Microsoft SharePoint 2016 on an ECS instance	Microsoft SharePoint Portal Server (Microsoft SharePoint) is a portal development environment that allows enterprises to develop intelligent portals. Microsoft SharePoint can be integrated with knowledge bases so that individual users and teams can connect to the environment. Microsoft SharePoint empowers your business by streamlining information processing.
Drupal content management framework	• Build a Drupal website based on an Alibaba Cloud Marketplace image • Manually build a Drupal website	Drupal is a free and open source content management framework (CMF) written in PHP. Drupal consists of a content management system (CMS) and a PHP development framework. If you are familiar with ECS and Linux and want to build websites on ECS instances, you can use images to build Drupal environments.

Website type	Deployment method	Description
Docker (CentOS 7)	Deploy and use Docker on Alibaba Cloud Linux 2 instances	Docker is an open source tool that allows you to encapsulate web applications in lightweight and portable standalone containers. Docker can run in almost all service environments. Docker is suitable for developers that are familiar with Linux but new to ECS instances.
LAMP environment	Build a LAMP environment	LAMP is a group of open source software programs used to build dynamic websites or servers. LAMP components include Linux, Apache, MySQL, and PHP. These components are all independent programs, but they are used together to form a powerful web application platform.
Common databases (Oracle, MySQL, and SQL Server)	Database overview	If you are familiar with ECS and MySQL databases, you can manually deploy MySQL databases to meet your requirements.
RabbitMQ	Deploy RabbitMQ	RabbitMQ is an open source implementation of Advanced Message Queuing Protocol (AMQP) that supports multiple clients, such as Python, Ruby, NET, Java, JMS, C, PHP, ActionScript, XMPP, STOMP, and AJAX. RabbitMQ is used to store and forward messages in distributed systems and is characterized by ease of use, scalability, and high availability.
Primary/secondary PostgreSQL architecture	Build a primary/secondary PostgreSQL system	ApsaraDB RDS for PostgreSQL is characterized by compatibility with NoSQL databases, efficient queries, plug-in management, high security, and excellent stability. If you are familiar with ECS, Linux, and PostgreSQL, you can manually deploy the primary/secondary PostgreSQL architecture.
SVN	Overview	Subversion (SVN) is an open source version control system used to manage ever-changing data.
Ghost blogging platform on CentOS 7	Build the Ghost blogging platform	Ghost is a free and open source blogging platform that is written in JavaScript and based on Node.js. The platform is used to simplify the online publishing process for personal bloggers and publishers. As your business expands, you can use the comprehensive services of Alibaba Cloud to scale up and scale out your business capacity.
FTP website	Manually build an FTP site on a Windows instance	You can build FTP servers in Windows for file storage and access.

Website type	Deployment method	Description
FTP website	Manually build an FTP site on a CentOS 7 instance	vsftpd is a light, safe, and easy-to-use FTP server for Linux. You can install vsftpd on ECS instances.

3.Build a software development environment

3.1. Deploy LNMP

3.1.1. Use ROS to deploy an LNMP environment

LNMP is an acronym of the names of its original four open source components: the Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. This topic describes how to use Alibaba Cloud Resource Orchestration Service (ROS) to deploy an LNMP environment.

Prerequisites

The first time that you use ROS, you are prompted to activate it. ROS is a free service. You can activate ROS free of charge.

Context

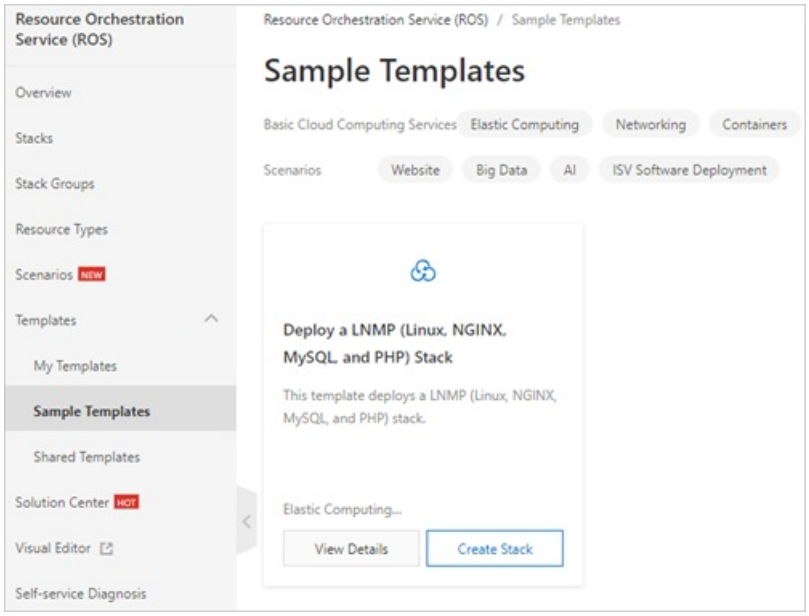
You do not need to download or install anything to use ROS. You can use ROS to create stack templates in the JSON format. In the ROS console, you can also use a sample template to create a stack. For more information, visit the Sample Templates page of the ROS console.

You can also use the sample templates provided by ROS to build environments, such as Java web test environments, Node.js development and test environments, Ruby web development and test environments, or Hadoop and Spark distributed systems. This topic uses the **Deploy a LNMP (Linux, NGINX, MySQL, and PHP) Stack** template to demonstrate how to use ROS to create an Elastic Compute Service (ECS) instance and deploy an LNMP environment on the instance.

For more information about ROS, see ROS documentation.

Procedure

1. Log on to the ROS console.
2. In the left-side navigation pane, choose **Templates > Sample Templates**.
3.
4. Find the **Deploy a LNMP (Linux, NGINX, MySQL, and PHP) Stack** template.



5. Click **View Details** to view the template in the JSON format.

   The following table describes the top-level fields of the JSON file.

Top-level field	Description
`"ROSTemplateFormatVersion": "2015-09-01"`	The version of the template. A value of 2015-09-01 is used.
`"Parameters": {...}`	Some parameters of the template. In this example, this field specifies the image ID, instance type, software download URLs, and software configurations, and default values are accepted for some of these parameters.
`"Resources": {...}`	The Alibaba Cloud resources that you can use the template to create. In this example, this field specifies that the resources to be created include an ECS instance and a security group. The properties of these resources are defined in the `Parameters` field.
`"Outputs": {...}`	The resource information that the stack generates after the specified resources are created. In this example, the stack generates the IP address that is used to access NGINX.
`"Description": "Deploy LNMP(Linux+Nginx+MySQL+PHP) stack on 1 ECS instance. *** WARNING *** Only support CentOS-7."`	The description of the template.

Top-level field	Description
`"Metadata": {...}`	Divides parameters in the Parameters field into groups. You can set labels for each group. In this example, parameters in the Parameters field are put into different groups based on whether they are ECS instance-related or software-related.

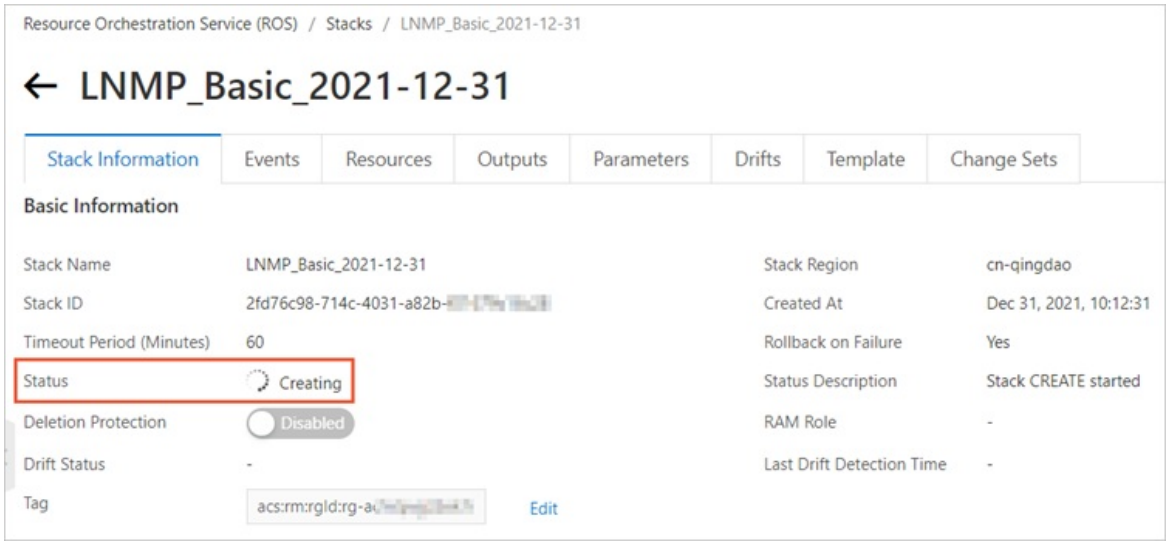
> **Note** For more information about sample templates of ROS, see Template structure.

6. In the upper-left corner of the page, click **Create Stack**.
7. Configure the parameters described in the following table and click **Create**.

Parameter	Description
Stack Name	The name of the stack.
Available Zone ID	(Required) The ID of the zone in which to create the ECS instance.
Image ID	The ID of the image to be used by the ECS instance.
Instance Type	(Required) The instance type of the ECS instance.
System Disk Category	The system disk category of the ECS instance.
Instance Password	(Required) The logon password of the ECS instance.
DB Name	The name of the MySQL database.
DB Username	The username of the MySQL database.
DB Password	(Required) The password of the MySQL database.
DB Root Password	(Required) The administrator (root) password of the MySQL database.
Nginx Source	The URL from which to download NGINX. We recommend that you use the default value.

> **Note** You can optionally click **Next** to go to the **Configure Stack (Optional)** step and then to the **Check and Confirm (Optional)** step. For more information, see Create a stack. In this example, default values are accepted for parameters in the Configure Stack (Optional) and Check and Confirm (Optional) steps.

After the stack is created, you are redirected to the details page of the stack. You can view the state of the stack on the Stack Information tab.



8. Click the **Outputs** tab and view the URL in the `Value` column corresponding to NginxWebsiteURL.

   You can use the URL to connect to the LNMP environment that you have created.

> **Note**
>
> - On the **Resources** tab, you can view all of the resources in the stack.
> - On the **Events** tab, you can view logs about the operations that ROS performed to create the stack.

3.1.2. Manually deploy an LNMP environment on an ECS instance that runs Alibaba Cloud Linux 2

NGINX is a small and efficient web server software that can be used to build an LNMP web service environment. LNMP is an acronym of the names of its original four open source components: Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. This topic describes how to manually build an LNMP environment on an Elastic Compute Service (ECS) instance that runs Alibaba Cloud Linux 2.1903 LTS 64-bit.

Prerequisites

- An ECS instance is created and a public IP address is assigned to the instance. For more information, see Creation method overview.

In this example, an ECS instance with the following configurations is used. We recommend that you do not change the operating system during deployment. Otherwise, errors may be reported when commands are run.

- Instance type: ecs.c6.large
- Operating system: Alibaba Cloud Linux 2.1903 LTS 64-bit
- Network type: Virtual Private Cloud (VPC)
- IP address: a public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on port 80. For more information, see Add a security group rule.

> **Note** For security purposes, this topic describes only the ports on which traffic must be allowed to deploy and test an LNMP environment. You can configure security group rules to allow traffic on more ports based on your needs. For example, if you want to connect to a MySQL database on an ECS instance, you must configure an inbound rule in a security group of the instance to allow traffic on port 3306, which is the default port used for MySQL.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites.

You can also purchase an LNMP image in Alibaba Cloud Marketplace and create an ECS instance from the image to build websites.

The following software versions are used in the sample procedure. If your software version differs from the preceding ones, you may need to adjust the commands and parameter settings.

- NGINX 1.20.1
- MySQL 5.7.36
- PHP 7.0.33

Step 1: Prepare the compilation environment

1. Connect to the ECS instance on which you want to deploy an LNMP environment.

   For more information, see Connection methodsGuidelines on instance connection.

2. Disable the firewall.

   i. Run the **systemctl status firewalld** command to check the state of the firewall.

```
[root@test ~]# systemctl status firewalld
   firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
eset: enabled)
   Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
     Docs: man:firewalld(1)
 Main PID: 20785 (firewalld)
```

   - If the firewall is in the *inactive* state, the firewall is disabled.
   - If the firewall is in the *active* state, the firewall is enabled.

ii. Disable the firewall. Skip this step if the firewall is already disabled.

- To temporarily disable the firewall, run the following command:

```
systemctl stop firewalld
```

> **Note** After you run this command, the firewall is temporarily disabled. When you restart the Linux instance, the firewall is enabled automatically.

- To permanently disable the firewall, perform the following steps:

a. Run the following command to disable the firewall:

```
systemctl stop firewalld
```

b. Run the following command to prevent the firewall from being automatically enabled at system startup:

```
systemctl disable firewalld
```

> **Note** You can re-enable the firewall after it is disabled. For more information, visit the official firewalld website.

3. Disable Security-Enhanced Linux (SELinux).

i. Run the **getenforce** command to check the state of SELinux.

- If SELinux is in the `Disabled` state, SELinux is disabled.
- If SELinux is in the `Enforcing` state, SELinux is enabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

You can disable SELinux on a temporary or permanent basis depending on your business needs. For more information, see Enable or disable SELinux.

Step 2: Install NGINX

> **Note** This topic provides the installation method for a single version of NGINX. If you want to install other versions of NGINX, see the "FAQ" section of the FAQ topic.

1. Run the following command to install NGINX:

```
yum -y install nginx
```

2. Run the following command to check the version of NGINX:

```
nginx -v
```

The following command output indicates that NGINX is installed:

```
nginx version: nginx/1.20.1
```

Step 3: Install MySQL

1. Run the following command to update the YUM repository:

```
rpm -Uvh  http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

2. Run the following command to install MySQL:

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

3. Run the following command to check the version of MySQL:

```
mysql -V
```

The following command output indicates that MySQL is installed:

```
mysql  Ver 14.14 Distrib 5.7.36, for Linux (x86_64) using  EditLine wrapper
```

4. Run the following command to start MySQL:

```
systemctl start mysqld
```

5. Run the following commands in sequence to enable automatic MySQL startup at system startup:

```
systemctl enable mysqld
systemctl daemon-reload
```

Step 4: Install PHP

1. Update the YUM repositories.
   i. Run the following commands to add the EPEL repository:

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

   ii. Run the following command to add the Webtatic repository:

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

2. Run the following command to install PHP:

```
yum -y install php70w-devel php70w.x86_64 php70w-cli.x86_64 php70w-common.x86_64 php70w-gd.x86_64 php70w-ldap.x86_64 php70w-mbstring.x86_64 php70w-mcrypt.x86_64  php70w-pdo.x86_64   php70w-mysqlnd  php70w-fpm php70w-opcache php70w-pecl-redis php70w-pecl-mongodb
```

3. Run the following command to check the version of PHP:

```
php -v
```

The following command output indicates that PHP is installed:

```
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
```

Step 5: Configure NGINX

1. Run the following command to back up the NGINX configuration file:

```
cp /etc/nginx/nginx.conf /etc/nginx/nginx.conf.bak
```

2. Modify the NGINX configuration file to add NGINX support for PHP.

> **Note** If you do not add this support, PHP-based pages cannot be displayed when you access them by using a browser.

i. Run the following command to open the NGINX configuration file:

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

ii. Press the /key to enter the edit mode.

iii. Modify or add the following information enclosed inside the `server` braces:

Retain the default values for all settings except the following:

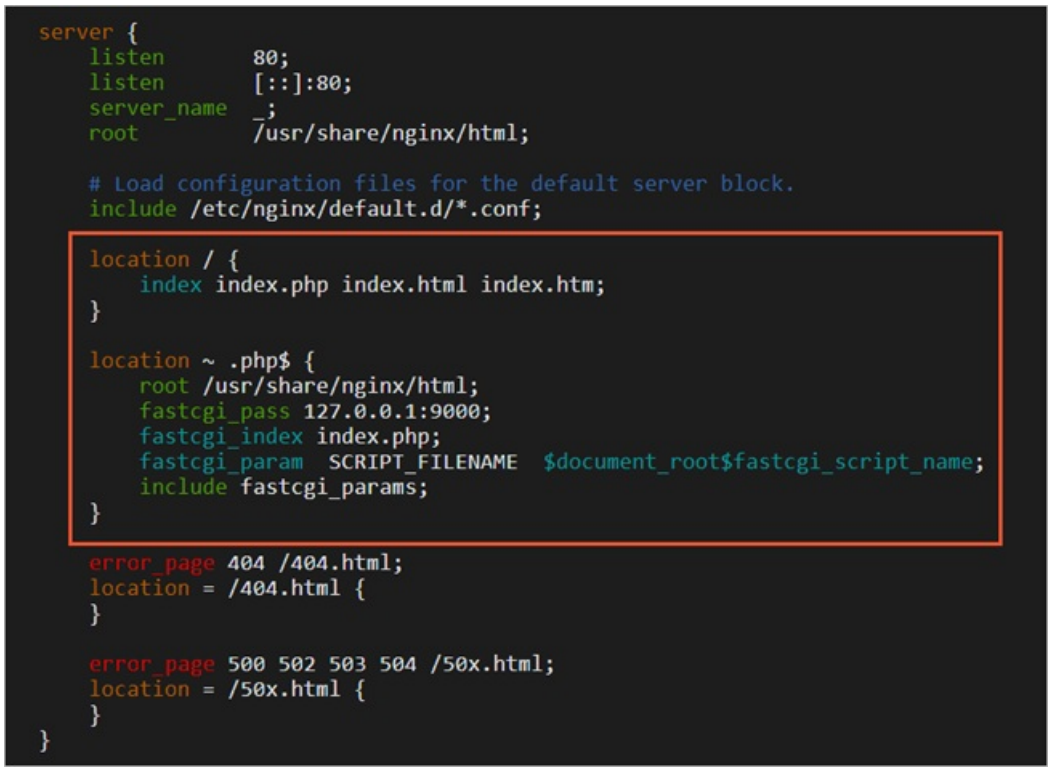
- Modify or add information enclosed inside the `location /` braces.

```
        location / {
            index index.php index.html index.htm;
        }
```

- Modify or add information enclosed inside the `location ~ .php$` braces.

```
        # Add the following information to make NGINX use Fast Common Gateway Int
erface (FastCGI) to process your PHP requests:
        location ~ .php$ {
            root /usr/share/nginx/html;    # Replace /usr/share/nginx/html with y
our website root directory. In this example, /usr/share/nginx/html is used as the
website root directory.
            fastcgi_pass 127.0.0.1:9000;   # NGINX forwards your PHP requests to
PHP FastCGI Process Manager (PHP-FPM) by using port 9000 of the ECS instance.
            fastcgi_index index.php;
            fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
            include fastcgi_params;   # NGINX calls the FastCGI operation to proc
ess the PHP requests.
        }
```

The following figure shows the added or modified configuration information.



```
server {
    listen       80;
    listen       [::]:80;
    server_name  _;
    root         /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    location / {
        index index.php index.html index.htm;
    }

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

    error_page 404 /404.html;
    location = /404.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}
```

iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.

3. Run the following command to start the NGINX service:

```
systemctl start nginx
```

4. Run the following command to enable automatic NGINX startup at system startup:

```
systemctl enable nginx
```

Step 6: Configure MySQL

1. Run the following command to check the */var/log/mysqld.log* file, and obtain and record the initial password of the root user:

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

The following command output is displayed. `ARQTRy3+n8*W` is the initial password of the root user. This initial password will be used when you reset the password of the root user.

```
2021-11-10T07:01:26.595215Z 1 [Note] A temporary password is generated for root@localho
st: ARQTRy3+n8*W
```

2. Run the following command to perform security configurations for MySQL:

```
mysql_secure_installation
```

i. Enter the initial password of the root user.

> Note

```
Securing the MySQL server deployment.
Enter password for user root: # Enter the initial password that you obtained in the preceding step.
```

ii. Reset the password of the root user.

```
The existing password for the user account root has expired. Please set a new password.
New password: # Enter a new password. The password must be 8 to 30 characters in length, and must contain uppercase letters, lowercase letters, digits, and special characters. Special characters include ( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /.
Re-enter new password: # Enter the new password again.
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 # The strength of the new password is contained in the command output.
Change the password for root ? (Press y|Y for Yes, any other key for No) : Y # Enter Y to confirm the new password.
# After the new password is set, you need to verify it again.
New password:# Enter the new password.
Re-enter new password: # Enter the new password again.
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 # Enter Y to confirm the new password.
```

iii. Enter *Y* to delete the anonymous user account.

```
Remove anonymous users? (Press y|Y for Yes, any other key for No) :Y
Success.
```

iv. Enter *Y* to deny remote access by the root user.

```
Disallow root login remotely? (Press y|Y for Yes, any other key for No) :Y
Success.
```

v. Enter *Y* to delete the test database and the access permissions on the database.

```
Remove test database and access to it? (Press y|Y for Yes, any other key for No) :Y
 - Dropping test database...
Success.
 - Removing privileges on test database...
Success.
```

vi. Enter *Y* to reload privilege tables.

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) :Y
Success.
All done!
```

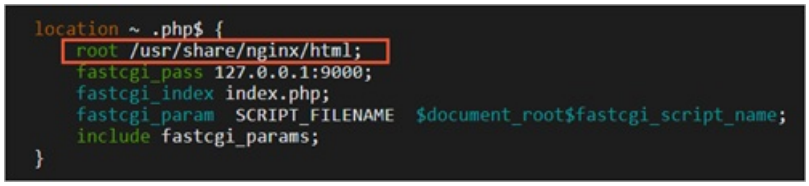
For more information, see the official MySQL documentation.

Step 7: Configure PHP

1. Create and edit the *phpinfo.php* file to show PHP information.
   i. Run the following command to create the *phpinfo.php* file:

   ```
   vim <website root directory> /phpinfo.php
   ```

   The *<website root directory>* is the `root` value enclosed inside the `location ~ .php$` braces that you configured in the *nginx.conf* file, as shown in the following figure.

   In this example, the website root directory is */usr/share/nginx/html*. You can run the following command to create the *phpinfo.php* file:

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

   ii. Press the /key to enter the edit mode.
   iii. Enter the following content. The `phpinfo()` function shows all configuration information of PHP.

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

   iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.
2. Run the following command to start PHP-FPM:

   ```
   systemctl start php-fpm
   ```

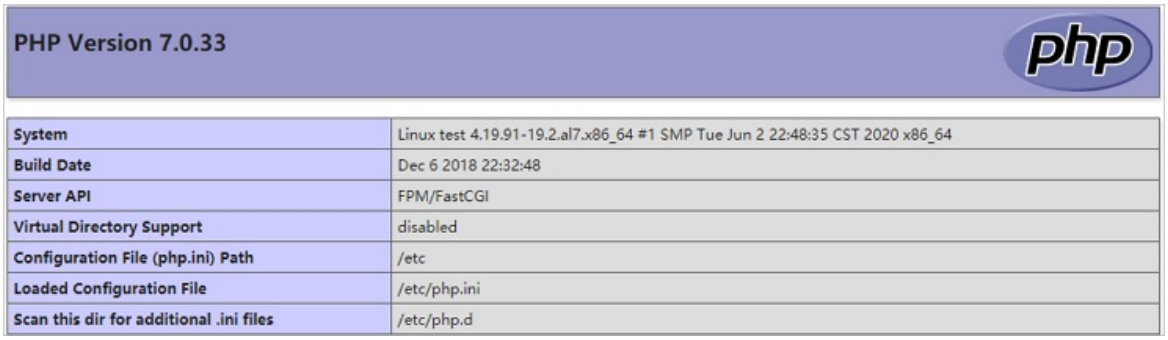
3. Run the following command to enable automatic PHP-FPM startup at system startup:

   ```
   systemctl enable php-fpm
   ```

Step 8: Test the connection to the LNMP environment

1. Open a browser on your Windows computer or another Windows host that can access the Internet.
2. In the address bar, enter `http://<Public IP address of the ECS instance>/phpinfo.php`.

   The following command output indicates that the LNMP environment is deployed.



PHP Version 7.0.33

System	Linux test 4.19.91-19.2.al7.x86_64 #1 SMP Tue Jun 2 22:48:35 CST 2020 x86_64
Build Date	Dec 6 2018 22:32:48
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

What's next

After you confirm that the LNMP environment is deployed, we recommend that you run the following command to delete the *phpinfo.php* file to ensure system security:

```
rm -rf <website root directory> /phpinfo.php
```

Replace the *<website root directory>* with the website root directory that you configured in the *nginx.conf* file.

In this example, the website root directory is */usr/share/nginx/html*. Run the following command:

```
rm -rf /usr/share/nginx/html/phpinfo.php
```

3.1.3. Manually build an LNMP environment on an Ubuntu 20 instance

NGINX is a small and efficient web server that can be used to build an LNMP web service environment. LNMP is an acronym of the names of its original four open source components: the Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. This topic describes how to manually build an LNMP environment on an Elastic Compute Service (ECS) instance that runs an Ubuntu 20 operating system.

Prerequisites

- An ECS instance is created and assigned a public IP address. For more information, see Create an instance by using the wizard.

  In this topic, an ECS instance that has the following configurations is used. To prevent command errors caused by operating system version issues, we recommend that you use the same operating system version as that used in this topic.

  - Instance type: ecs.c6.large
  - Operating system: Ubuntu 20.04
  - Network type: Virtual Private Cloud (VPC)
  - IP address: public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on ports 22, 80, and 443. For more information, see Add a security group rule.

> **Note** For security purposes, this topic describes only the ports on which traffic must be allowed to build and test an LNMP environment. You can configure security group rules to allow traffic on more ports based on your needs. For example, if you want to connect to a MySQL database on an ECS instance, you must add an inbound rule to a security group of the instance to allow traffic on port 3306, which is the default port used for MySQL.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites.

The following software versions are used in the sample procedure:

- NGINX 1.18.0
- MySQL 8.0.27
- PHP 7.4.3

Step 1: Make preparations

1. Connect to the ECS instance on which you want to build an LNMP environment.

   For more information, see Connection methodsGuidelines on instance connection.
2. Disable the firewall on the instance operating system.
   i. Run the following command to check the state of the firewall:

   ```
   sudo ufw status
   ```

   - If the firewall is disabled and in the inactive state, `Status: inactive` is displayed.
   - If the firewall is enabled and in the active state, `Status: active` is displayed.

   ii. (Optional) Disable the firewall.

   If the firewall is enabled, run the following command to disable the firewall and prevent the firewall from starting on instance startup:

   ```
   sudo ufw disable
   ```

   > **Note** If you want to re-enable the firewall after it is disabled and start the firewall on instance startup, run the **sudo ufw enable** command.

Step 2: Install NGINX

1. Run the following command to update software packages in the Ubuntu operating system:

   ```
   sudo apt update
   ```
2. Run the following command to install NGINX:

   ```
   sudo apt -y install nginx
   ```
3. Run the following command to check the version of NGINX:

   ```
   sudo nginx -v
   ```

The following command output indicates that NGINX is installed and its version is 1.18.0.

```
nginx version: nginx/1.18.0 (Ubuntu)
```

Step 3: Install MySQL

1. Run the following command to install MySQL:

   ```
   sudo apt -y install mysql-server
   ```

2. Run the following command to check the version of MySQL:

   ```
   sudo mysql -V
   ```

   The following command output indicates that MySQL is installed and its version is 8.0.27.

   ```
   mysql  Ver 8.0.27-0ubuntu0.20.04.1 for Linux on x86_64 ((Ubuntu))
   ```

Step 4: Install PHP

1. Run the following command to install PHP:

   ```
   sudo apt -y install php-fpm
   ```

2. Run the following command to check the version of PHP:

   ```
   sudo php -v
   ```

   The following command output indicates that PHP is installed and its version is 7.4.3.

   ```
   PHP 7.4.3 (cli) (built: Nov 25 2021 23:16:22) ( NTS )
   Copyright (c) The PHP Group
   Zend Engine v3.4.0, Copyright (c) Zend Technologies
       with Zend OPcache v7.4.3, Copyright (c), by Zend Technologies
   ```

Step 5: Configure NGINX

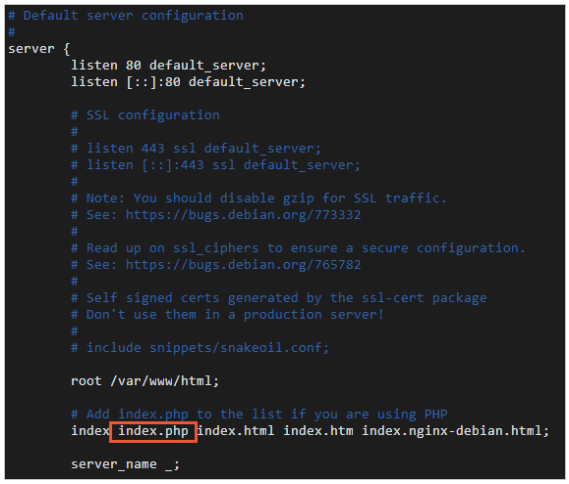
1. Run the following commands to modify the default NGINX configuration file and add NGINX support for PHP to the file.

   i. Open the default NGINX configuration file.

   ```
   sudo vim /etc/nginx/sites-enabled/default
   ```

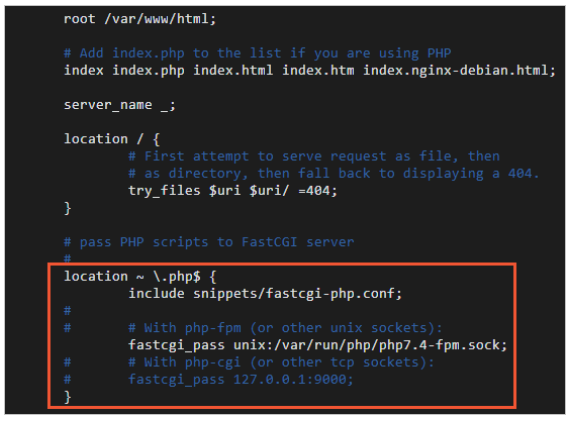
ii. Press the /key to enter the edit mode to modify the file.

a. Find the configuration line that starts with `index` within the `server` braces and add `index.php` to the line.



b. Find `location ~ \.php$ {}` within the `server` braces and delete the annotation character (#) from the following configuration lines within the location ~ \.php$ braces:

```
location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
}
```



iii. Press the *Esc* key to exit the edit mode. Then, enter `:wq` and press the *Enter* key to save and close the file.

2. Run the following command to restart NGINX:

```
sudo systemctl restart nginx.service
```

Step 6: Configure MySQL

1. Run the following command to configure the security settings of MySQL:

```
sudo mysql_secure_installation
```

2. Follow the command line instructions to configure the following settings in sequence.

i. Enter *Y* to use the password authentication tool that comes with MySQL.

```
VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?
Press y|Y for Yes, any other key for No: Y
```

ii. Specify the password strength.

In this example, enter `1` to use the `MEDIUM` password strength. You can use the password strength that your business requires. We recommend that you use a strong password to improve data protection.

```
There are three levels of password validation policy:
LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
file
Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 1
```

iii. Set the MySQL password.

> **Note**

```
Please set the password for root here.
New password:
Re-enter new password:
Estimated strength of the password: 100
```

iv. Enter *Y* to confirm to use the set password.

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

v. Enter *Y* to delete the autonomous user that comes with MySQL.

```
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
```

vi. Enter *Y* to deny connection to MySQL by the root user.

```
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
```

vii. Enter *Y* to delete the test database and the access permissions on the database.

```
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
```

viii. Enter *Y* to reload privilege tables.

```
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
```

If `All done!` is displayed in the command output, the configuration is complete.

3. Check whether you can log on to the MySQL database.
   i. Run the following command to log on to MySQL:

```
sudo mysql -uroot -p
```

   ii. Enter the password you set for MySQL after the `Enter password:` prompt.

> **Note**
>
> The following command output indicates that you are logged on to MySQL.

```
root@test:~# sudo mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

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

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

mysql>
```

   iii. Run the following command to exit MySQL:

```
exit;
```

Step 7: Configure PHP

1. Run the following command to create the *phpinfo.php* file in the NGINX website root directory:

```
sudo vi <Website root directory>/phpinfo.php
```

<Website root directory> is a variable, which can be viewed in the NGINX configuration file. In this topic, the NGINX configuration file is the default */etc/nginx/sites-enabled/default* file. You can run the **cat /etc/nginx/sites-enabled/default** command to view the file content. The following command output shows that the website root directory of NGINX is `/var/www/html`.

```
# Default server configuration
#
server {
        listen 80 default_server;
        listen [::]:80 default_server;

        # SSL configuration
        #
        # listen 443 ssl default_server;
        # listen [::]:443 ssl default_server;
        #
        # Note: You should disable gzip for SSL traffic.
        # See: https://bugs.debian.org/773332
        #
        # Read up on ssl_ciphers to ensure a secure configuration.
        # See: https://bugs.debian.org/765782
        #
        # Self signed certs generated by the ssl-cert package
        # Don't use them in a production server!
        #
        # include snippets/snakeoil.conf;

        root /var/www/html;

        # Add index.php to the list if you are using PHP
        index index.php index.html index.htm index.nginx-debian.html;
```

Run the following command to create the phpinfo.php file in the /var/www/html directory:

```
sudo vi /var/www/html/phpinfo.php
```

2. Press the /key to enter the edit mode and add the following configuration to the phpinfo.php file:

   Invoke the `phpinfo()` function to show all configuration information of PHP.

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

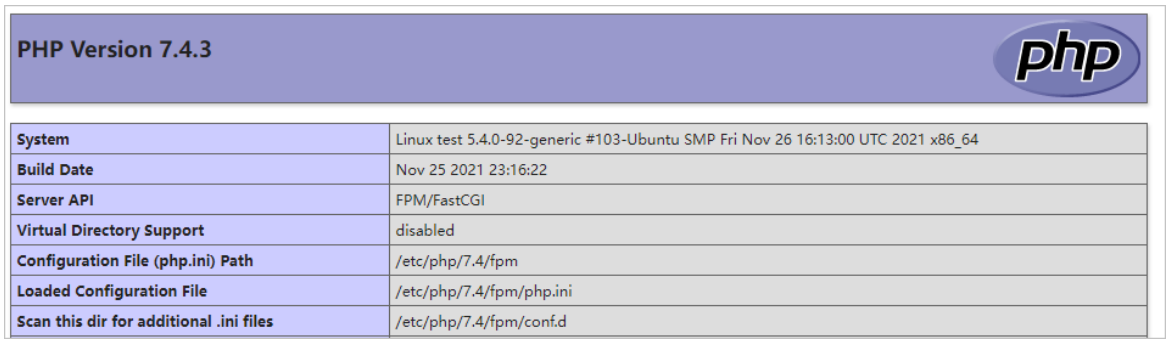
3. Press the *Esc* key to exit the edit mode. Then, enter `:wq` and press the *Enter* key to save and close the file.
4. Run the following command to start PHP:

   ```
   sudo systemctl start php7.4-fpm
   ```

Step 8: Test the connection to the PHP configuration page

1. Open a browser on your Windows computer or another Windows host that can access the Internet.
2. In the address bar, enter `http://<Public IP address of the ECS instance>/phpinfo.php`.

   The following page shows the PHP configuration page, which indicates that the LNMP environment is built.



PHP Version 7.4.3

System	Linux test 5.4.0-92-generic #103-Ubuntu SMP Fri Nov 26 16:13:00 UTC 2021 x86_64
Build Date	Nov 25 2021 23:16:22
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/fpm
Loaded Configuration File	/etc/php/7.4/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/fpm/conf.d

What's next

After the LNMP environment is built, we recommend that you delete the *phpinfo.php* test file to prevent data leaks.

```
rm -rf <Website root directory> /phpinfo.php
```

In this topic, the website root directory is `/var/www/html`. Run the following command to delete the test file:

```
rm -rf /var/www/html/phpinfo.php
```

3.1.4. Manually build an LNMP environment on a CentOS 8 instance

LNMP is an acronym of the names of its original four open source components: the Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. This topic describes how to manually build an LNMP environment on an Elastic Compute Service (ECS) instance that runs a CentOS 8 operating system.

Prerequisites

- An ECS instance is created and a public IP address is assigned to the instance. For more information, see Creation method overview.

  In this example, an ECS instance with the following configurations is used. We recommend that you do not change the operating system during deployment. Otherwise, errors may be reported when commands are run.

  - Instance type: ecs.c6.large
  - Operating system: CentOS 7.8 64-bit public image
  - Network type: Virtual Private Cloud (VPC)
  - IP address: a public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on ports 22, 80, and 443. For more information, see Add a security group rule.

  > **Note** For security purposes, this topic describes only the ports on which traffic must be allowed to deploy and test an LNMP environment. You can configure security group rules to allow traffic on more ports based on your needs. For example, if you want to connect to a MySQL database on an ECS instance, you must configure an inbound rule in a security group of the instance to allow traffic on port 3306, which is the default port used for MySQL.

Context

By default, the Dandified YUM (DNF) package manager is installed in CentOS 8. DNF is the next-generation version of Yellowdog Updater Modified (YUM). You can run the **dnf** command in CentOS 8 to obtain related instructions.

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites.

You can also purchase an LNMP image in Alibaba Cloud Marketplace and create an ECS instance from the image to build websites.

In the topic, an ECS instance that has the following configurations is used. Operations may vary based on the configurations of your instance.

- Instance type: ecs.c6.large

- Operating system: CentOS 8.1 64-bit public image
- CPU: 2 vCPUs
- Memory: 4 GiB
- Network type: Virtual Private Cloud (VPC)
- IP address: public IP address

The following software versions are used. If you use software versions different from the following ones, you may need to adjust commands and parameter settings.

- NGINX 1.16.1
- MySQL 8.0.17
- PHP 7.3.5

Step 1: Prepare the compilation environment

1. Connect to the ECS instance on which you want to deploy an LNMP environment.

   For more information, see Connection methodsGuidelines on instance connection.

2. Disable the firewall.

   i. Run the **systemctl status firewalld** command to check the state of the firewall.

   ```
   [root@test ~]# systemctl status firewalld
      firewalld.service - firewalld - dynamic firewall daemon
      Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
   eset: enabled)
      Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
        Docs: man:firewalld(1)
    Main PID: 20785 (firewalld)
   ```

   - If the firewall is in the *inactive* state, the firewall is disabled.
   - If the firewall is in the *active* state, the firewall is enabled.

   ii. Disable the firewall. Skip this step if the firewall is already disabled.

   - To temporarily disable the firewall, run the following command:

     ```
     systemctl stop firewalld
     ```

     Note After you run this command, the firewall is temporarily disabled. When you restart the Linux instance, the firewall is enabled automatically.

   - To permanently disable the firewall, perform the following steps:

     a. Run the following command to disable the firewall:

     ```
     systemctl stop firewalld
     ```

     b. Run the following command to prevent the firewall from being automatically enabled at system startup:

     ```
     systemctl disable firewalld
     ```

     Note You can re-enable the firewall after it is disabled. For more information, visit the official firewalld website.

3. Disable Security-Enhanced Linux (SELinux).
   i. Run the **getenforce** command to check the state of SELinux.
      - If SELinux is in the `Disabled` state, SELinux is disabled.
      - If SELinux is in the `Enforcing` state, SELinux is enabled.
   ii. Disable SELinux. Skip this step if SELinux is already disabled.

      You can disable SELinux on a temporary or permanent basis depending on your business needs. For more information, see Enable or disable SELinux.

Step 2: Install NGINX

1. Change the CentOS 8 repository address.
2. Run the following command to install NGINX.

   In this example, NGINX 1.16.1 is used.

   > **Note** Go to the official NGINX website to view the list of NGINX packages suited for CentOS 8.

   ```
   dnf -y install http://nginx.org/packages/centos/8/x86_64/RPMS/nginx-1.16.1-1.el8.ngx.x86_64.rpm
   ```

3. Run the following command to check theNGINX version:

   ```
   nginx -v
   ```

   The following command output shows the NGINX version:

   ```
   nginx version: nginx/1.16.1
   ```

Step 3: Install MySQL

1. Run the following command to install MySQL:

   ```
   dnf -y install @mysql
   ```

2. Run the following command to check the MySQL version:

   ```
   mysql -V
   ```

   The following command output shows the NGINX version:

   ```
   mysql  Ver 8.0.17 for Linux on x86_64 (Source distribution)
   ```

Step 4: Install PHP

1. Run the following commands to add and update the Extra Packages for Enterprise Linux (EPEL) repository:

   ```
   dnf -y install epel-release
   dnf update epel-release
   ```

2. Run the following commands to delete unneeded cached software packages and update the software repository:

```
dnf clean all
dnf makecache
```

3. Run the following command to start the `php:7.3` module.

> **Note** In this example, `PHP 7.3` is used. If you want to use `PHP 7.4`, you must install the remi repository by running the **dnf -y install https://rpms.remirepo.net/enterprise/remi-release-8.rpm** command.

```
dnf module enable php:7.3
```

4. Run the following command to install the PHP modules:

```
dnf install php php-curl php-dom php-exif php-fileinfo php-fpm php-gd php-hash php-json
php-mbstring php-mysqli php-openssl php-pcre php-xml libsodium
```

5. Run the following command to check the PHP version:

```
php -v
```

The following command output shows the NGINX version:

```
PHP 7.3.5 (cli) (built: Apr 30 2019 08:37:17) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.3.5, Copyright (c) 1998-2018 Zend Technologies
```

Step 5: Configure NGINX

1. Run the following command to check the default path of the NGINX configuration file:

```
cat /etc/nginx/nginx.conf
```

The `include` configuration item within the `http` braces indicates the default path of the configuration file.

```
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;
}
```

2. Run the following commands to back up the default configuration file in the default path of the configuration file:

```
cd /etc/nginx/conf.d
cp default.conf default.conf.bak
```

3. Modify the default configuration file.
   i. Run the following command to open the default configuration file:

```
vi default.conf
```

   ii. Press the /key to enter the edit mode.
   iii. Make the following modifications to the content within the `location` braces:

```
location / {
    # Replace the path with the root directory of your website.
    root   /usr/share/nginx/html;
    # Add the default homepage index.php.
    index  index.html index.htm index.php;
}
```

   iv. Remove the annotation character `#` in front of `location ~ \.php$` and modify the content within the braces.

   The following modifications are made:

```
location ~ \.php$ {
    # Replace the path with the root directory of your website.
    root           /usr/share/nginx/html;
    NGINX is associated with PHP-FPM by using UNIX sockets. This configuration must
be the same as the listen configuration in the /etc/php-fpm.d/www.conf file.
    fastcgi_pass   unix:/run/php-fpm/www.sock;
    fastcgi_index  index.php;
    # Change /scripts$fastcgi_script_name to $document_root$fastcgi_script_name.
    fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
    # NGINX calls the FastCGI operation to process the PHP requests.
    include        fastcgi_params;
}
```

   > **Note** The following methods can be used for inter-process communication between NGINX and PHP-FPM.
   >
   > - Use TCP sockets: NGINX can communicate with PHP-FPM across instances over the network.
   > - Use UNIX domain sockets: NGINX can communicate with PHP-FPM only within a single instance without using the network.

   v. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the configuration file.

4. Run the following command to start NGINX:

```
systemctl start nginx
```

5. Run the following command to enable NGINX to start on instance startup:

```
systemctl enable nginx
```

Step 6: Configure MySQL

1. Run the following command to start MySQL and enable it to start on instance startup:

```
systemctl enable --now mysqld
```

2. Run the following command to check whether MySQL is started:

```
systemctl status mysqld
```

If MySQL is started, the command output contains `Active: active (running)`.

3. Run the following command to make security configurations for MySQL and set the password:

```
mysql_secure_installation
```

After you run the command, perform the following operations based on the command prompts:

i. Enter *Y* and press the Enter key to make the configurations.

ii. Enter *2* as the password strength and press the Enter key.

0 indicates a low password strength, *1* indicates a medium password strength, and *2* indicates a high password strength. We recommend that you select a high password strength.

iii. Enter a new password and confirm it.

In this example, the password is `PASSword123!`.

iv. Enter *Y* and press the Enter key to use the password.

v. Enter *Y* and press the Enter key to delete anonymous users.

vi. Set whether to allow remote access to MySQL.

- Enter *Y* and press the Enter key to deny remote access.
- Enter *N* or a key instead of *Y* and press the Enter key to allow remote access.

vii. Enter *Y* and press the Enter key to delete the `test` database and access permissions on the `test` database.

viii. Enter *Y* and press the Enter key to reload privilege tables.

Step 7: Configure PHP

1. Modify the PHP configuration file.

i. Run the following command to open the configuration file:

```
vi /etc/php-fpm.d/www.conf
```

ii. Press the /key to enter the edit mode.

iii. Find the `user = apache` and `group = apache` lines, and change `apache` to `nginx`.

```
; Unix user/group of processes
; Note: The user is mandatory. If the group is not set, the default user's group
;       will be used.
; RPM: apache user chosen to provide access to the same directories as httpd
user = nginx
; RPM: Keep a group allowed to write in log dir.
group = nginx
```

iv. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the configuration file.

2. Create and edit the *phpinfo.php* file to show PHP information.

i. Run the following command to create the *phpinfo.php* file:

```
vim <website root directory> /phpinfo.php
```

The *<website root directory>* is the `root` value enclosed inside the `location ~ .php$` braces that you configured in the *nginx.conf* file, as shown in the following figure.

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

In this example, the website root directory is */usr/share/nginx/html*. You can run the following command to create the *phpinfo.php* file:

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

ii. Press the /key to enter the edit mode.

iii. Enter the following content. The `phpinfo()` function shows all configuration information of PHP.

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

iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.

3. Run the following command to start `PHP-FPM`:

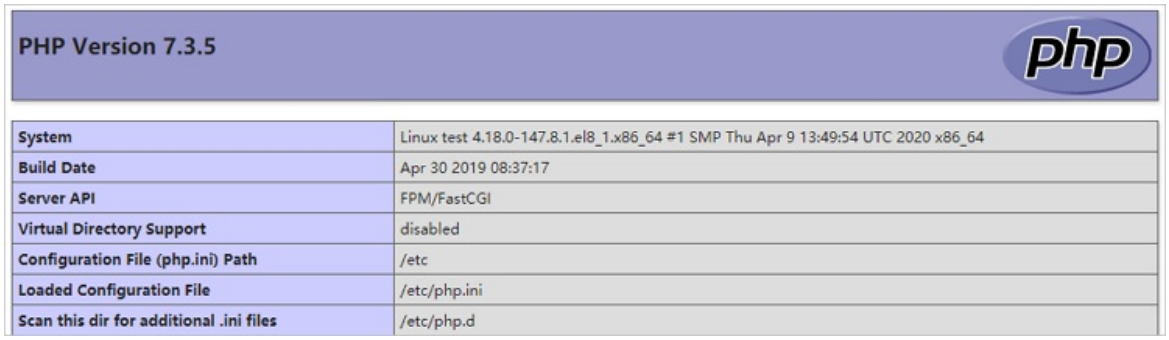
```
systemctl start php-fpm
```

4. Run the following command to enable `PHP-FPM` to start on instance startup:

```
systemctl enable php-fpm
```

Step 8: Test the connection to the LNMP environment

1. Open the browser on your computer.
2. In the address bar, enter `http://<Public IP address of the ECS instance>/phpinfo.php`. The following command output indicates that the LNMP environment is deployed.



What's next

After you confirm that the LNMP environment is deployed, we recommend that you run the following command to delete the *phpinfo.php* file to ensure system security:

```
rm -rf <website root directory> /phpinfo.php
```

Replace the *<website root directory>* with the website root directory that you configured in the *nginx.conf* file.

In this example, the website root directory is */usr/share/nginx/html*. Run the following command:

```
rm -rf /usr/share/nginx/html/phpinfo.php
```

3.1.5. Manually build an LNMP environment on a CentOS 7 instance

NGINX is a small and efficient web server software that can be used to build an LNMP web service environment. LNMP is an acronym of the names of its original four open source components: Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language. This topic describes how to manually build an LNMP environment on an Elastic Compute Service (ECS) instance that runs a CentOS 7 operating system.

Prerequisites

- An ECS instance is created and a public IP address is assigned to the instance. For more information, see Creation method overview.

  In this example, an ECS instance with the following configurations is used. We recommend that you do not change the operating system during deployment. Otherwise, errors may be reported when commands are run.

  - Instance type: ecs.c6.large
  - Operating system: CentOS 7.8 64-bit public image
  - Network type: Virtual Private Cloud (VPC)
  - IP address: a public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on ports 22, 80, and 443. For more information, see Add a security group rule.

> **Note** For security purposes, this topic describes only the ports on which traffic must be allowed to deploy and test an LNMP environment. You can configure security group rules to allow traffic on more ports based on your needs. For example, if you want to connect to a MySQL database on an ECS instance, you must configure an inbound rule in a security group of the instance to allow traffic on port 3306, which is the default port used for MySQL.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites.

You can also purchase an LNMP image in Alibaba Cloud Marketplace and create an ECS instance from the image to build websites.

The following software versions are used in the sample procedure. If your software version differs from the preceding ones, you may need to adjust the commands and parameter settings.

- NGINX 1.20.1
- MySQL 5.7.36
- PHP 7.0.33

Step 1: Prepare the compilation environment

1. Connect to the ECS instance on which you want to deploy an LNMP environment.

   For more information, see Connection methodsGuidelines on instance connection.

2. Disable the firewall.

   i. Run the **systemctl status firewalld** command to check the state of the firewall.

   ```
   [root@test ~]# systemctl status firewalld
      firewalld.service - firewalld - dynamic firewall daemon
      Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
   eset: enabled)
      Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
        Docs: man:firewalld(1)
    Main PID: 20785 (firewalld)
   ```

   - If the firewall is in the *inactive* state, the firewall is disabled.
   - If the firewall is in the *active* state, the firewall is enabled.

ii. Disable the firewall. Skip this step if the firewall is already disabled.

- To temporarily disable the firewall, run the following command:

```
systemctl stop firewalld
```

> **Note** After you run this command, the firewall is temporarily disabled. When you restart the Linux instance, the firewall is enabled automatically.

- To permanently disable the firewall, perform the following steps:

a. Run the following command to disable the firewall:

```
systemctl stop firewalld
```

b. Run the following command to prevent the firewall from being automatically enabled at system startup:

```
systemctl disable firewalld
```

> **Note** You can re-enable the firewall after it is disabled. For more information, visit the official firewalld website.

3. Disable Security-Enhanced Linux (SELinux).

i. Run the **getenforce** command to check the state of SELinux.

- If SELinux is in the `Disabled` state, SELinux is disabled.
- If SELinux is in the `Enforcing` state, SELinux is enabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

You can disable SELinux on a temporary or permanent basis depending on your business needs. For more information, see Enable or disable SELinux.

Step 2: Install NGINX

> **Note** This topic provides the installation method for a single version of NGINX. If you want to install other versions of NGINX, see the "FAQ" section in this topic.

1. Run the following command to install NGINX:

```
yum -y install nginx
```

2. Run the following command to check the version of NGINX:

```
nginx -v
```

The following command output indicates that NGINX is installed:

```
nginx version: nginx/1.20.1
```

Step 3: Install MySQL

1. Run the following command to update the YUM repository:

```
rpm -Uvh  http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

2. Run the following command to install MySQL:

> **Note** If you are using an operating system whose kernel version is el8, you may receive the `No match for argument` error message. If this occurs, run the **yum module disable mysql** command to disable the default MySQL module before you install MySQL.

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

3. Run the following command to check the version of MySQL:

```
mysql -V
```

The following command output indicates that MySQL is installed:

```
mysql  Ver 14.14 Distrib 5.7.36, for Linux (x86_64) using  EditLine wrapper
```

4. Run the following command to start MySQL:

```
systemctl start mysqld
```

5. Run the following commands in sequence to enable automatic MySQL startup at system startup:

```
systemctl enable mysqld
systemctl daemon-reload
```

Step 4: Install PHP

1. Update the YUM repositories.
   i. Run the following commands to add the EPEL repository:

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

   ii. Run the following command to add the Webtatic repository:

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

2. Run the following command to install PHP:

```
yum -y install php70w-devel php70w.x86_64 php70w-cli.x86_64 php70w-common.x86_64 php70w
-gd.x86_64 php70w-ldap.x86_64 php70w-mbstring.x86_64 php70w-mcrypt.x86_64  php70w-pdo.x
86_64   php70w-mysqlnd  php70w-fpm php70w-opcache php70w-pecl-redis php70w-pecl-mongodb
```

3. Run the following command to check the version of PHP:

```
php -v
```

The following command output indicates that PHP is installed:

```
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
```

Step 5: Configure NGINX

1. Run the following command to back up the NGINX configuration file:

```
cp /etc/nginx/nginx.conf /etc/nginx/nginx.conf.bak
```

2. Modify the NGINX configuration file to add NGINX support for PHP.

> **Note** If you do not add this support, PHP-based pages cannot be displayed when you access them by using a browser.

   i. Run the following command to open the NGINX configuration file:

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

   ii. Press the /key to enter the edit mode.

   iii. Modify or add the following information enclosed inside the `server` braces:

   Retain the default values for all settings except the following:

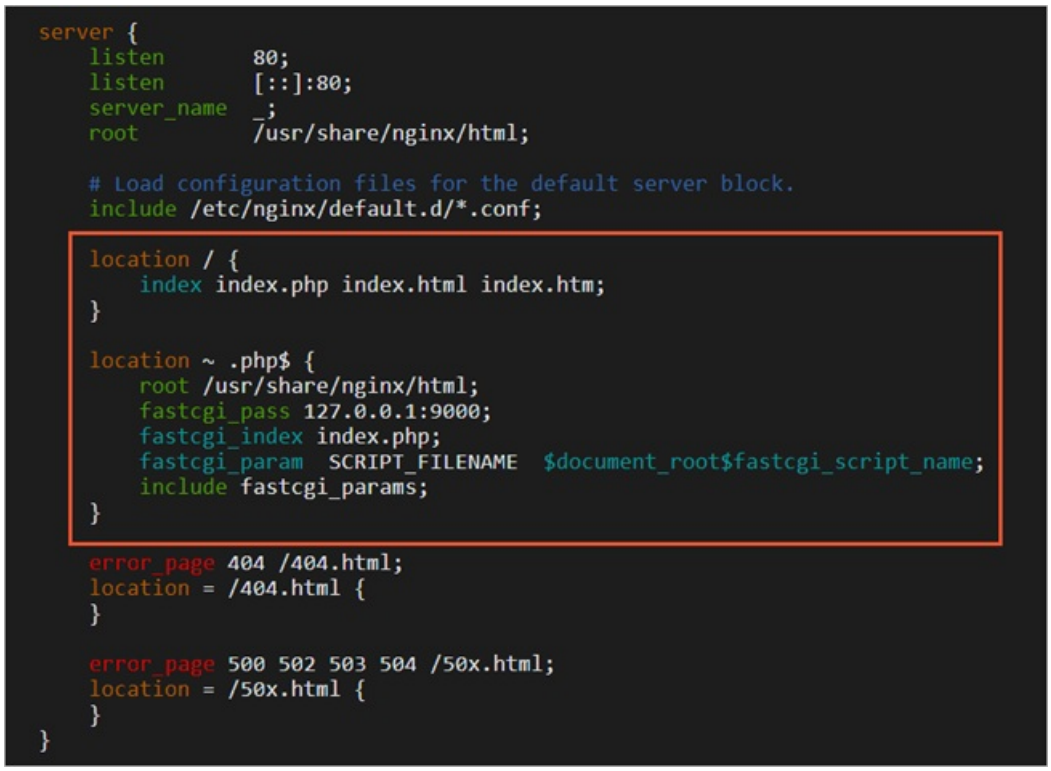
   - Modify or add information enclosed inside the `location /` braces.

```
        location / {
            index index.php index.html index.htm;
        }
```

   - Modify or add information enclosed inside the `location ~ .php$` braces.

```
        # Add the following information to make NGINX use Fast Common Gateway Int
erface (FastCGI) to process your PHP requests:
        location ~ .php$ {
            root /usr/share/nginx/html;    # Replace /usr/share/nginx/html with y
our website root directory. In this example, /usr/share/nginx/html is used as the
website root directory.
            fastcgi_pass 127.0.0.1:9000;   # NGINX forwards your PHP requests to
PHP FastCGI Process Manager (PHP-FPM) by using port 9000 of the ECS instance.
            fastcgi_index index.php;
            fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
            include fastcgi_params;   # NGINX calls the FastCGI operation to proc
ess the PHP requests.
        }
```

The following figure shows the added or modified configuration information.



```
server {
    listen       80;
    listen       [::]:80;
    server_name  _;
    root         /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    location / {
        index index.php index.html index.htm;
    }

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

    error_page 404 /404.html;
    location = /404.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}
```

iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.

3. Run the following command to start the NGINX service:

```
systemctl start nginx
```

4. Run the following command to enable automatic NGINX startup at system startup:

```
systemctl enable nginx
```

Step 6: Configure MySQL

1. Run the following command to check the */var/log/mysqld.log* file, and obtain and record the initial password of the root user:

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

The following command output is displayed. `ARQTRy3+n8*W` is the initial password of the root user. This initial password will be used when you reset the password of the root user.

```
2021-11-10T07:01:26.595215Z 1 [Note] A temporary password is generated for root@localho
st: ARQTRy3+n8*W
```

2. Run the following command to perform security configurations for MySQL:

```
mysql_secure_installation
```

i. Enter the initial password of the root user.

> Note

```
Securing the MySQL server deployment.
Enter password for user root: # Enter the initial password that you obtained in the preceding step.
```

ii. Reset the password of the root user.

```
The existing password for the user account root has expired. Please set a new password.
New password: # Enter a new password. The password must be 8 to 30 characters in length, and must contain uppercase letters, lowercase letters, digits, and special characters. Special characters include ( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /.
Re-enter new password: # Enter the new password again.
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 # The strength of the new password is contained in the command output.
Change the password for root ? (Press y|Y for Yes, any other key for No) : Y # Enter Y to confirm the new password.
# After the new password is set, you need to verify it again.
New password:# Enter the new password.
Re-enter new password: # Enter the new password again.
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 # Enter Y to confirm the new password.
```

iii. Enter *Y* to delete the anonymous user account.

```
Remove anonymous users? (Press y|Y for Yes, any other key for No) :Y
Success.
```

iv. Enter *Y* to deny remote access by the root user.

```
Disallow root login remotely? (Press y|Y for Yes, any other key for No) :Y
Success.
```

v. Enter *Y* to delete the test database and the access permissions on the database.

```
Remove test database and access to it? (Press y|Y for Yes, any other key for No) :Y
 - Dropping test database...
Success.
 - Removing privileges on test database...
Success.
```

vi. Enter *Y* to reload privilege tables.

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) :Y
Success.
All done!
```

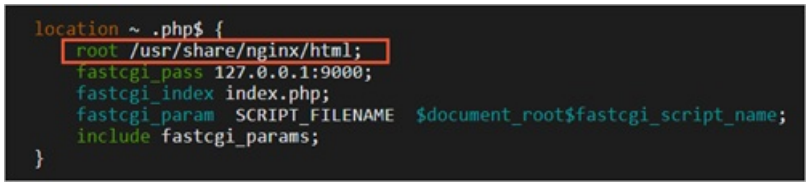
For more information, see the official MySQL documentation.

Step 7: Configure PHP

1. Create and edit the *phpinfo.php* file to show PHP information.
   i. Run the following command to create the *phpinfo.php* file:

   ```
   vim <website root directory> /phpinfo.php
   ```

   The *<website root directory>* is the `root` value enclosed inside the `location ~ .php$` braces that you configured in the *nginx.conf* file, as shown in the following figure.

   In this example, the website root directory is */usr/share/nginx/html*. You can run the following command to create the *phpinfo.php* file:

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

   ii. Press the /key to enter the edit mode.
   iii. Enter the following content. The `phpinfo()` function shows all configuration information of PHP.

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

   iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.
2. Run the following command to start PHP-FPM:

   ```
   systemctl start php-fpm
   ```

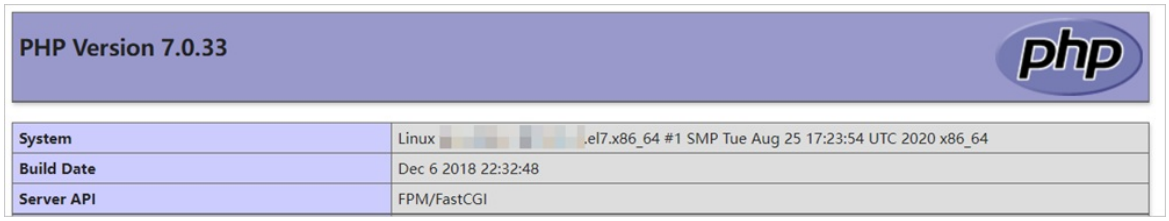
3. Run the following command to enable automatic PHP-FPM startup at system startup:

   ```
   systemctl enable php-fpm
   ```

Step 8: Test the connection to the LNMP environment

1. Open a browser on your Windows computer or another Windows host that can access the Internet.
2. In the address bar, enter `http://<public IP address of the ECS instance>/phpinfo.php`.

   The following page indicates that the LNMP environment is deployed.

What to do next

After you confirm that the LNMP environment is deployed, we recommend that you run the following command to delete the *phpinfo.php* file to ensure system security:

```
rm -rf <website root directory> /phpinfo.php
```

Replace the *<website root directory>* with the website root directory that you configured in the *nginx.conf* file.

In this example, the website root directory is */usr/share/nginx/html*. Run the following command:

```
rm -rf /usr/share/nginx/html/phpinfo.php
```

FAQ

How do I install other NGINX versions?

1. Use a browser to visit the NGINX open source community to obtain the download URLs of NGINX versions.

   Select the NGINX version that you want to install. NGINX 1.8.1 is used in this example.

2. Connect to the ECS instance on which you want to deploy an LNMP environment.
3. Run the **wget** command to download NGINX 1.8.1.

   You can obtain the URL of the NGINX installation package for the required version from the NGINX open source community. Then, run the `wget URL` command to download the NGINX installation package to the ECS instance. For example, you can download NGINX 1.8.1 by running the following command:

   ```
   wget http://nginx.org/download/nginx-1.8.1.tar.gz
   ```

4. Run the following command to install NGINX dependencies:

   ```
   yum install -y gcc-c++
   yum install -y pcre pcre-devel
   yum install -y zlib zlib-devel
   yum install -y openssl openssl-devel
   ```

5. Run the following command to decompress the NGINX 1.8.1 installation package. Then, go to the folder in which NGINX resides:

   ```
   tar zxvf nginx-1.8.1.tar.gz
   cd nginx-1.8.1
   ```

6. Run the following commands in sequence to compile the source code:

   ```
   ./configure \
    --user=nobody \
    --group=nobody \
    --prefix=/usr/local/nginx \
    --with-http_stub_status_module \
    --with-http_gzip_static_module \
    --with-http_realip_module \
    --with-http_sub_module \
    --with-http_ssl_module
   ```

```
make && make install
```

7. Run the following command to go to the sbin directory of NGINX, and then start NGINX:

```
cd /usr/local/nginx/sbin/
./nginx
```

8. Use a browser to access `<public IP address of the ECS instance>`.

   If the following page appears, it indicates that NGINX was installed successfully and started.

3.1.6. Manually build an LNMP environment on a CentOS 6 instance

This topic describes how to manually build an LNMP environment on an Elastic Compute Service (ECS) instance that runs a CentOS 6 operating system. LNMP is an acronym of the names of its original four open source components: Linux operating system, NGINX web server, MySQL relational database management system, and PHP programming language.

Prerequisites

- An inbound rule is added to a security group of the ECS instance to allow traffic on port 80. For more information, see Add a security group rule.
- The CentOS 6 source address is changed. For more information, see Change the CentOS 6 source address.

> **Note** CentOS 6 has reached its end of life (EOL). The CentOS 6 source has been removed from the `http://mirror.centos.org/centos-6/` source address in compliance with the CentOS community rules. You must manually change the CentOS 6 source address to ensure that the YUM repository is available. Otherwise, errors are reported when you run **yum** commands.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites.

This topic describes how to manually build an LNMP environment. You can also purchase an LNMP image in Alibaba Cloud Marketplace and create an ECS instance from the image to build websites.

The following configurations and software versions are used in the example:

- Instance type: ecs.c6.large
- Operating system: a CentOS 6.8 32-bit public image

> **Note** If you are using a 32-bit operating system, select an instance type with up to 4 GiB of memory.

- NGINX: nginx 1.10.2
- MySQL: MySQL 5.6.24
- PHP: PHP 5.6.23
- Network type: Virtual Private Cloud (VPC)
- IP address: a public IP address

If you use software versions different from the preceding ones, you may need to adjust commands and parameter settings.

Step 1: Prepare the compilation environment

1. Create a CentOS 6 instance.

   For more information, see Create an instance by using the wizard.
2. Connect to the CentOS 6 instance.

   For more information, see Connect to a Linux instance by using a password.
3. Run the `cat /etc/redhat-release` command to check the operating system version.

   ```
   [root@test ~]# cat /etc/redhat-release
   CentOS release 6.8 (Final)
   ```

Step 2: Install and configure NGINX

1. Run the following commands in sequence to add a user to run the NGINX service process:

   ```
   groupadd -r nginx
   ```

   ```
   useradd -r -g nginx nginx
   ```
2. Download the source code package and then decompress and compile the package.
   i. Run the following command to download the source code package:

   ```
   wget http://nginx.org/download/nginx-1.10.2.tar.gz
   ```

   ii. Run the following command to decompress the source code package:

   ```
   tar xvf nginx-1.10.2.tar.gz -C /usr/local/src
   ```

   iii. Run the following commands in sequence to install compilation tools:

   ```
   yum groupinstall "Development tools"
   ```

   ```
   yum -y install gcc wget gcc-c++ automake autoconf libtool libxml2-devel libxslt-devel perl-devel perl-ExtUtils-Embed pcre-devel openssl-devel
   ```

   iv. Run the following command to go to the directory of the NGINX source code package:

   ```
   cd /usr/local/src/nginx-1.10.2
   ```

v. Run the following commands to compile the source code:

```
./configure \
--prefix=/usr/local/nginx \
--sbin-path=/usr/sbin/nginx \
--conf-path=/etc/nginx/nginx.conf \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--pid-path=/var/run/nginx.pid \
--lock-path=/var/run/nginx.lock \
--http-client-body-temp-path=/var/tmp/nginx/client \
--http-proxy-temp-path=/var/tmp/nginx/proxy \
--http-fastcgi-temp-path=/var/tmp/nginx/fcgi \
--http-uwsgi-temp-path=/var/tmp/nginx/uwsgi \
--http-scgi-temp-path=/var/tmp/nginx/scgi \
--user=nginx \
--group=nginx \
--with-pcre \
--with-http_v2_module \
--with-http_ssl_module \
--with-http_realip_module \
--with-http_addition_module \
--with-http_sub_module \
--with-http_dav_module \
--with-http_flv_module \
--with-http_mp4_module \
--with-http_gunzip_module \
--with-http_gzip_static_module \
--with-http_random_index_module \
--with-http_secure_link_module \
--with-http_stub_status_module \
--with-http_auth_request_module \
--with-mail \
--with-mail_ssl_module \
--with-file-aio \
--with-ipv6 \
--with-http_v2_module \
--with-threads \
--with-stream \
--with-stream_ssl_module
```

```
make && make install
```

vi. Run the following command to create a directory:

```
mkdir -p /var/tmp/nginx/client
```

3. Add the SysV startup script.

i. Run the `vi /etc/init.d/nginx` command to open the SysV startup script file.

ii. Press the /key and add the following content to the script file:

```
#!/bin/sh
#
# nginx - this script starts and stops the nginx daemon
```

```
#
# chkconfig:   - 85 15
# description: Nginx is an HTTP(S) server, HTTP(S) reverse \
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# config:      /etc/sysconfig/nginx
# pidfile:     /var/run/nginx.pid
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0
nginx="/usr/sbin/nginx"
prog=$(basename $nginx)
NGINX_CONF_FILE="/etc/nginx/nginx.conf"
[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx
lockfile=/var/lock/subsys/nginx
start() {
    [ -x $nginx ] || exit 5
    [ -f $NGINX_CONF_FILE ] || exit 6
    echo -n $"Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}
stop() {
    echo -n $"Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
killall -9 nginx
}
restart() {
    configtest || return $?
    stop
    sleep 1
    start
}
reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
RETVAL=$?
    echo
}
force_reload() {
    restart
}
```

```
configtest() {
$nginx -t -c $NGINX_CONF_FILE
}
rh_status() {
    status $prog
}
rh_status_q() {
    rh_status >/dev/null 2>&1
}
case "$1" in
    start)
        rh_status_q && exit 0
    $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload
        ;;
    status)
        rh_status
        ;;
    condrestart|try-restart)
        rh_status_q || exit 0
            ;;
    *)
      echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|fo
rce-reload|configtest}"
        exit 2
esac
```

iii. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the SysV startup script file.

4. Run the following command to grant execute permissions on the script:

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

5. Run the following commands in sequence to add NGINX to the list of system services and enable NGINX to start on system startup:

```
chkconfig --add nginx
```

```
chkconfig  nginx on
```

6. Run the following command to start the NGINX service:

```
service nginx start
```

7. Test whether NGINX is installed.
   i. Log on to the ECS console.
   ii. In the left-side navigation pane, choose **Instances & Images > Instances**.
   iii. On the **Instances** page, find the instance that you created and copy its public IP address from the **IP Address** column.
   iv. In the browser address bar, enter the IP address and press the Enter key.
   The following page indicates that NGINX is installed.

Step 3: Install and configure MySQL

1. Run the following commands in sequence to prepare the compilation environment:

```
yum groupinstall "Server Platform Development"  "Development tools" -y
```

```
yum install cmake -y
```

2. Create a directory to store MySQL data.
   i. Run the `mkdir /mnt/data` command to create a directory to store MySQL data.
   ii. Run the `groupadd -r mysql` command to create a user group named `mysql`.
   iii. Run the `useradd -r -g mysql -s /sbin/nologin mysql` command to create a user named `mysql`.
   iv. Run the `id mysql` command to check whether the user is created.
   v. Run the `chown -R mysql:mysql /mnt/data` command to change the group and user of the MySQL data directory to `mysql`.
3. Download the latest stable version of the source code package and then decompress and compile it.
   i. Run one of the following commands to download the source code package:

```
wget https://dev.mysql.com/get/Downloads/mysql-5.6.24.tar.gz
```

```
wget https://cdn.mysql.com/archives/mysql-5.6/mysql-5.6.24.tar.gz
```

   ii. Run the following command to decompress the source code package:

```
tar xvf mysql-5.6.24.tar.gz -C  /usr/local/src
```

iii. Run the following command to go to the directory of the MySQL source code package:

```
cd /usr/local/src/mysql-5.6.24
```

iv. Run the following commands in sequence to compile the source code package:

```
cmake . -DCMAKE_INSTALL_PREFIX=/usr/local/mysql \
-DMYSQL_DATADIR=/mnt/data \
-DSYSCONFDIR=/etc \
-DWITH_INNOBASE_STORAGE_ENGINE=1 \
-DWITH_ARCHIVE_STORAGE_ENGINE=1 \
-DWITH_BLACKHOLE_STORAGE_ENGINE=1 \
-DWITH_READLINE=1 \
-DWITH_SSL=system \
-DWITH_ZLIB=system \
-DWITH_LIBWRAP=0 \
-DMYSQL_TCP_PORT=3306 \
-DMYSQL_UNIX_ADDR=/tmp/mysql.sock \
-DDEFAULT_CHARSET=utf8 \
-DDEFAULT_COLLATION=utf8_general_ci
```

```
make && make install
```

4. Configure MySQL.

i. Run the following command to change the group and user of the MySQL installation directory to mysql:

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

ii. Run the following commands in sequence to initialize the database:

```
cd /usr/local/mysql
```

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

> **Note** After MySQL is installed on CentOS 6.8, a file named *my.cnf* appears in the */etc* directory. You must change the file name. For example, change the file name to */etc/my.cnf.bak*. Otherwise, the file interferes with the configuration process of MySQL and causes MySQL to fail to start.

iii. Run the following commands in sequence to copy the configuration file of MySQL:

```
cp /usr/local/mysql/support-files/mysql.server /etc/init.d/mysqld
```

```
cp /usr/local/mysql/support-files/my-default.cnf /etc/my.cnf
```

iv. Run the following command to grant execute permissions on the startup script of MySQL:

```
chmod +x /etc/init.d/mysqld
```

v. Run the following commands in sequence to add MySQL to the list of system services and enable MySQL to start on system startup:

```
chkconfig --add mysqld
```

```
chkconfig mysqld  on
```

vi. Run the following command to change the installation and data storage paths in the configuration file:

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

vii. Run the following commands to set the PATH environment variable:

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

```
source /etc/profile.d/mysql.sh
```

5. Run the following command to start the MySQL service:

```
service mysqld start
```

6. Run the following command to connect to the MySQL database for testing:

```
mysql -h 127.0.0.1
```

Step 4: Install PHP-FPM

NGINX acts as a web server and does not directly call or parse external programs when it receives requests. It must use Fast Common Gateway Interface (FastCGI) to call external programs. However, in case of a PHP request, NGINX transfers the request to a PHP interpreter and returns the result to the client. PHP-FPM is a FastCGI process manager that can parse PHP code. PHP-FPM provides better PHP process management methods to effectively control memory and processes and smoothly reload PHP configurations.

1. Run the following command to install the dependency:

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

2. Download the latest stable version of the source code package, and decompress and compile it.

i. Run the following command to download the source code package:

```
wget http://cn2.php.net/get/php-5.6.23.tar.bz2/from/this/mirror
```

> Note

ii. Run the following commands in sequence to decompress the source code package:

```
cp mirror php-5.6.23.tar.bz2
```

```
tar xvf php-5.6.23.tar.bz2 -C /usr/local/src
```

iii. Run the following command to go to the directory of the PHP source code package:

```
cd /usr/local/src/php-5.6.23
```

iv. Run the following commands in sequence to compile the source code package:

```
./configure --prefix=/usr/local/php \
--with-config-file-scan-dir=/etc/php.d \
--with-config-file-path=/etc \
--with-mysql=/usr/local/mysql \
--with-mysqli=/usr/local/mysql/bin/mysql_config \
--enable-mbstring \
--with-freetype-dir \
--with-jpeg-dir \
--with-png-dir \
--with-zlib \
--with-libxml-dir=/usr \
--with-openssl \
--enable-xml \
--enable-sockets \
--enable-fpm \
--with-mcrypt \
--with-bz2
```

```
make && make install
```

3. Configure PHP.

i. Run the following commands in sequence to add the PHP and PHP-FPM configuration files:

```
cp /usr/local/src/php-5.6.23/php.ini-production /etc/php.ini
```

```
cd /usr/local/php/etc/
```

```
cp php-fpm.conf.default php-fpm.conf
```

```
sed -i 's@;pid = run/php-fpm.pid@pid = /usr/local/php/var/run/php-fpm.pid@' php-fpm
.conf
```

ii. Run the following command to add the PHP-FPM startup script:

```
cp /usr/local/src/php-5.6.23/sapi/fpm/init.d.php-fpm /etc/init.d/php-fpm
```

iii. Run the following command to grant execute permissions on the PHP-FPM startup script:

```
chmod +x /etc/init.d/php-fpm
```

iv. Run the following commands in sequence to add PHP-FPM to the list of system services and enable PHP-FPM to start on system startup:

```
chkconfig --add php-fpm
```

```
chkconfig --list php-fpm
```

```
chkconfig php-fpm on
```

4. Run the following command to start PHP-FPM:

```
service php-fpm start
```

5. Add the NGINX support for FastCGI.
   i. Run the following command to back up the default NGINX configuration file:

```
cp /etc/nginx/nginx.conf /etc/nginx/nginx.conf.bak
```

   ii. Run the following command to add the NGINX configuration file:

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

   iii. Run the `vi /etc/nginx/nginx.conf` command to open the NGINX configuration file.
   iv. Press the /key and add index.php to the index line to support .php index page files.

   Example:

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

   v. Delete the annotation symbol in front of the following content:

```
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;
}
```

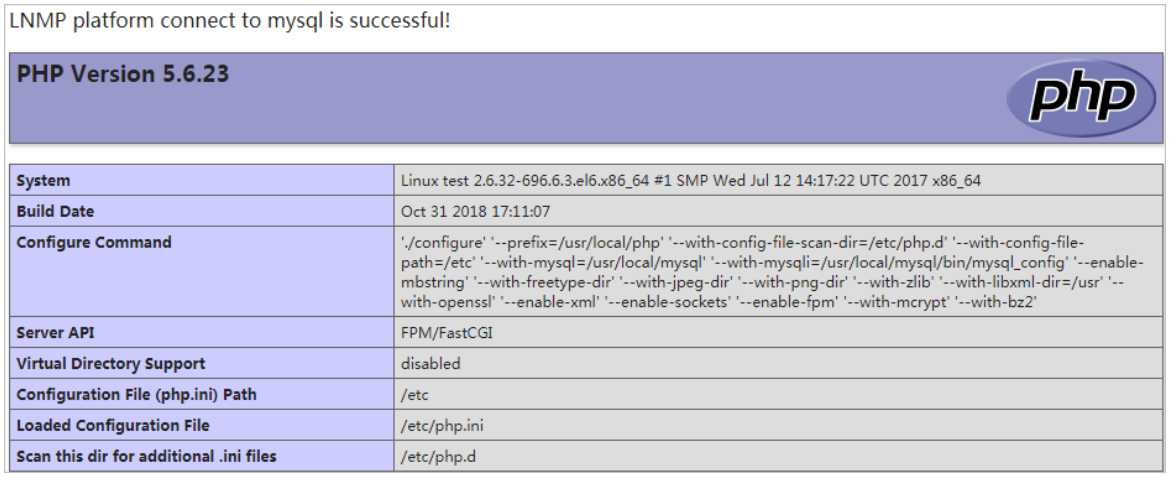
   vi. Change `root html;` to `root /usr/local/nginx/html;`.
   vii. Change `fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;` to `fastcgi_param SCRIPT_FILENAME /usr/local/nginx/html/$fastcgi_script_name;`.
   viii. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the NGINX configuration file.
6. Run the `service nginx reload` command to reload the NGINX configuration file.
7. Modify the *index.php* file.
   i. Run the `vi /usr/local/nginx/html/index.php` command to open the *index.php* file.
   ii. Press the /key and enter the following content:

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

iii. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the *index.php* file.

Step 5: Test the connection to the LNMP environment

1.
2.
3. On the **Instances** page, find the instance where the LNMP environment is built and copy the public IP address of the instance from the **IP Address** column.
4. In the browser address bar, enter the IP address and press the Enter key.
   The following page indicates that the LNMP environment is built.


LNMP platform connect to mysql is successful!

PHP Version 5.6.23

System	Linux test 2.6.32-696.6.3.el6.x86_64 #1 SMP Wed Jul 12 14:17:22 UTC 2017 x86_64
Build Date	Oct 31 2018 17:11:07
Configure Command	'./configure' '--prefix=/usr/local/php' '--with-config-file-scan-dir=/etc/php.d' '--with-config-file-path=/etc' '--with-mysql=/usr/local/mysql' '--with-mysqli=/usr/local/mysql/bin/mysql_config' '--enable-mbstring' '--with-freetype-dir' '--with-jpeg-dir' '--with-png-dir' '--with-zlib' '--with-libxml-dir=/usr' '--with-openssl' '--enable-xml' '--enable-sockets' '--enable-fpm' '--with-mcrypt' '--with-bz2'
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d

3.2. Build a LAMP environment

3.2.1. Build a LAMP environment on a Ubuntu 20 instance

LAMP is an acronym of the names of its original four components: the Linux operating system, Apache HTTP Server, MySQL relational database management system, and PHP programming language. LAMP environments are commonly used web development environments. This topic describes how to build a LAMP environment on an Elastic Compute Service (ECS) instance that runs Ubuntu.

Prerequisites

- An ECS instance is created and assigned a public IP address. For more information, see Create an instance by using the wizard.

  In this example, an ECS instance that has the following configurations is used. We recommend that you use the same operating system version as that used in the example to prevent command errors caused by operating system version issues.

  - Instance type: ecs.c6.large
  - Operating system: Ubuntu 20.04 64-bit public image
  - Network type: Virtual Private Cloud (VPC)
  - IP address: a public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on ports 22 and 80.

For more information, see Add a security group rule.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites. The following software versions are used in the sample procedure. The operations may vary based on your instance type and software versions.

- Apache: 2.4.41
- MySQL: 8.0.27
- PHP: 7.4.3

Step 1: Make preparations

1. Connect to the ECS instance on which you want to build a LAMP environment.

   For more information, see Connection methodsGuidelines on instance connection.

2. Disable the firewall on the instance operating system.

   i. Run the following command to check the state of the firewall:

   ```
   sudo ufw status
   ```

   - If the firewall is disabled and in the inactive state, `Status: inactive` is displayed.
   - If the firewall is enabled and in the active state, `Status: active` is displayed.

   ii. (Optional) Disable the firewall.

   If the firewall is enabled, run the following command to disable the firewall and prevent the firewall from starting on instance startup:

   ```
   sudo ufw disable
   ```

   > **Note** If you want to re-enable the firewall after it is disabled and start the firewall on instance startup, run the **sudo ufw enable** command.

Step 2: Install Apache

1. Run the following command to update the Ubuntu software packages:

   ```
   sudo apt update
   ```

2. Run the following command to install Apache:

   ```
   sudo apt-get -y install apache2
   ```

3. Run the following command to check the Apache version:

   ```
   apache2 -v
   ```

   The following example command output indicates that Apache is installed and its version is 2.4.41:

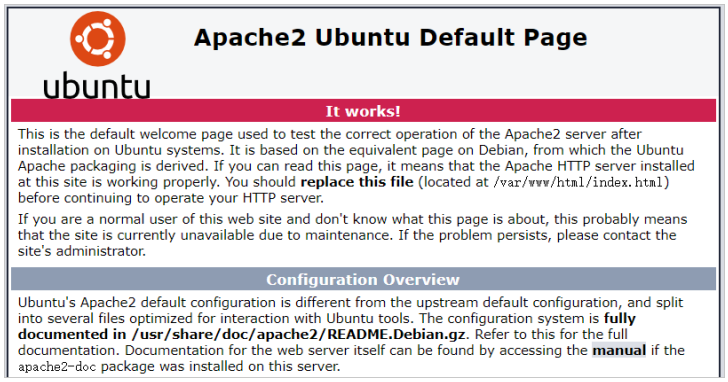
   ```
   Server version: Apache/2.4.41 (Ubuntu)
   Server built:   2022-01-05T14:49:56
   ```

4. Run the following command to start Apache and configure Apache to start on system startup:

```
sudo systemctl start apache2
```

5. On your Windows computer or another host that can access the Internet, use a browser to access <*Public IP address of the ECS instance*>.

   If the following default Apache homepage is displayed, Apache is working normally.



Step 3: Install and configure MySQL

1. Install MySQL.
   i. Run the following command to install MySQL:

   ```
   sudo apt -y install mysql-server
   ```

   ii. Run the following command to check the MySQL version:

   ```
   sudo mysql -V
   ```

   The following example command output indicates that MySQL is installed and its version is 8.0.27:

   ```
   mysql  Ver 8.0.27-0ubuntu0.20.04.1 for Linux on x86_64 ((Ubuntu))
   ```

2. Run the following command to start MySQL:

   ```
   sudo systemctl start mysql
   ```

3. Configure MySQL.
   i. Run the following command to configure the security settings of MySQL:

   ```
   sudo mysql_secure_installation
   ```

   ii. Enter *Y* to use the password verification tool that comes with MySQL.

   ```
   VALIDATE PASSWORD COMPONENT can be used to test passwords
   and improve security. It checks the strength of password
   and allows the users to set only those passwords which are
   secure enough. Would you like to setup VALIDATE PASSWORD component?
   Press y|Y for Yes, any other key for No: Y
   ```

iii. Configure a password strength.

In this example, `1` is used, which indicates the `MEDIUM` password strength. You can configure a password strength based on your business requirements. We recommend that you use the STRONG password strength to improve data protection.

```
There are three levels of password validation policy:
LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
file
Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 1
```

iv. Set a password for MySQL.

> Note

```
Please set the password for root here.
New password:
Re-enter new password:
Estimated strength of the password: 100
```

v. Enter *Y* to use the password that you set.

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

vi. Enter *Y* to delete the anonymous user account that comes with MySQL.

```
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
```

vii. Enter *Y* to deny remote access by the root account to MySQL.

```
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
```

viii. Enter *Y* to delete the test database and the access permissions on the database in MySQL.

```
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
```

ix. Enter *Y* to reload privilege tables.

```
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
```

When the configuration is complete, `All done!` is displayed in the command output.

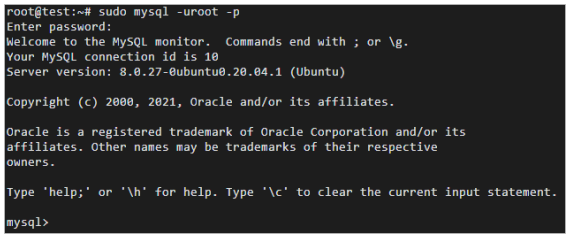
4. Check whether you can log on to MySQL.
   i. Run the following command to log on to MySQL:

   ```
   sudo mysql -uroot -p
   ```

   ii. At the `Enter password:` prompt, enter the password that you set for MySQL.

   > **Note**

   The following example command output indicates that you are logged on to MySQL.

   iii. Run the following command to exit MySQL:

   ```
   exit;
   ```

Step 4: Install PHP

1. Run the following command to install PHP:

   ```
   sudo apt -y install php-fpm
   ```

2. Run the following command to check the PHP version:

   ```
   sudo php -v
   ```

   The following example command output indicates that PHP is installed and its version is 7.4.3:

   ```
   PHP 7.4.3 (cli) (built: Nov 25 2021 23:16:22) ( NTS )
   Copyright (c) The PHP Group
   Zend Engine v3.4.0, Copyright (c) Zend Technologies
       with Zend OPcache v7.4.3, Copyright (c), by Zend Technologies
   ```

Step 5: Create and access a test webpage

1. In the root directory of the Apache website, create a test webpage.

i. Run the following command to view the path information of the root directory of the Apache website:

```
sudo cat /etc/apache2/sites-available/000-default.conf
```

The `DocumentRoot /var/www/html` line in the command output indicates that the website root directory is */var/www/html*.

ii. Run the following command to create a test webpage in the website root directory and add the `phpinfo()` functions to the webpage.

The `phpinfo()` function is used to show all configuration information of PHP.

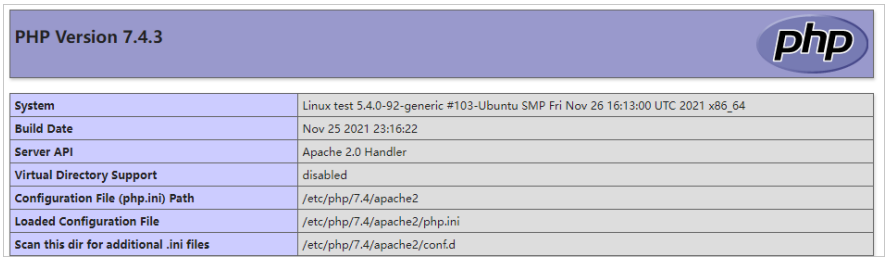
```
sudo echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

2. Run the following command to restart Apache:

```
sudo systemctl restart apache2
```

3. On your Windows computer or another host that can access the Internet, use a browser to access `<Public IP address of the ECS instance>/phpinfo.php`.

The following page shows PHP settings and indicates that the LAMP environment is built.



What's next

After the LAMP environment is built, we recommend that you delete the *phpinfo.php* test file to prevent data leaks.

```
rm -rf <Website root directory> /phpinfo.php
```

In this example, `/var/www/html` is used as the website root directory. Run the following command to delete the test file:

```
rm -rf /var/www/html/phpinfo.php
```

3.2.2. Build a LAMP environment on a CentOS 7 instance

This topic describes how to build a LAMP stack on an Elastic Compute Service (ECS) instance. LAMP is an acronym of the names of its original four components: the Linux operating system, Apache HTTP Server, MySQL relational database management system, and PHP programming language.

Prerequisites

- An ECS instance is created and assigned a public IP address. For more information, see Creation

method overview.

In this example, an ECS instance that has the following configurations is used. We recommend that you use the same operating system version as that used in the example to prevent command errors caused by operating system version issues.

- Instance type: ecs.c6.large
- Operating system: CentOS 7.8 64-bit public image
- Network type: Virtual Private Cloud (VPC)
- IP address: a public IP address

- An inbound rule is added to a security group of the ECS instance to allow traffic on ports 22 and 80. For more information, see Add a security group rule.

Context

This topic is intended for individual users who are familiar with Linux operating systems but new to using Alibaba Cloud ECS to build websites. The following software versions are used in the sample procedure. The operations may vary based on your instance type and software versions.

- Apache HTTP Server: 2.4.6
- MySQL: 5.7.31
- PHP 7.0.33
- phpMyAdmin 4.0.10.20

This topic describes how to manually build a LAMP stack. You can also purchase a LAMP image on Alibaba Cloud Marketplace and create an ECS instance from the image to build websites.

Step 1: Make preparations

1. Create an instance by using the wizard.
2. Connect to a Linux instance by using a password.
3. Run the **cat /etc/redhat-release** command to check the operating system version.

```
[root@test ~]# cat /etc/redhat-release
CentOS Linux release 7.8.2003 (Core)
```

4. Disable the firewall.
   i. Run the **systemctl status firewalld** command to check the state of the firewall.

```
[root@test ~]# systemctl status firewalld
   firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
eset: enabled)
   Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
     Docs: man:firewalld(1)
 Main PID: 20785 (firewalld)
```

   - If the firewall is in the *inactive* state, the firewall is disabled.
   - If the firewall is in the *active* state, the firewall is enabled. In this example, the firewall is in the active state and must be disabled.

ii. Disable the firewall. Skip this step if the firewall is already disabled.

- To temporarily disable the firewall, run the **systemctl stop firewalld** command.

> **Note** After you run the preceding command, the firewall is disabled. The next time you restart the Linux operating system, the firewall is re-enabled and enters the active state.

- To permanently disable the firewall, run the **systemctl disable firewalld** command.

> **Note** You can re-enable the firewall after it is disabled. For more information, visit the official firewalld website.

5. Disable Security-Enhanced Linux (SELinux).

i. Run the **getenforce** command to check the state of SELinux.

```
[root@test ~]# getenforce
Enforcing
```

- If SELinux is in the *Disabled* state, SELinux is disabled.
- If SELinux is in the *Enforcing* state, SELinux is enabled. In this example, SELinux is in the Enforcing state and must be disabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

- To temporarily disable SELinux, run the `setenforce 0` command.

> **Note** After you run the preceding command, SELinux is disabled. The next time you restart the Linux operating system, SELinux is enabled and enters the Enforcing state.

- To permanently disable SELinux, run the **vi /etc/selinux/config** command to edit the SELinux configuration file. Press the Enter key. Move the pointer to the `SELINUX=enforcing` line and press the /key to switch to the edit mode. Change the line to `SELINUX=disabled` and press the *Esc* key. Enter *:wq* and press the Enter key to save and close the SELinux configuration file. Restart the operating system to apply the settings.

> **Note** You can re-enable SELinux after it is disabled. For more information, see SELinux documentation.

Step 2: Install Apache

1. Run the following command to install Apache and its extension package:

```
yum -y install httpd httpd-manual mod_ssl mod_perl mod_auth_mysql
```

2. Run the **httpd -v** command to check the Apache version.

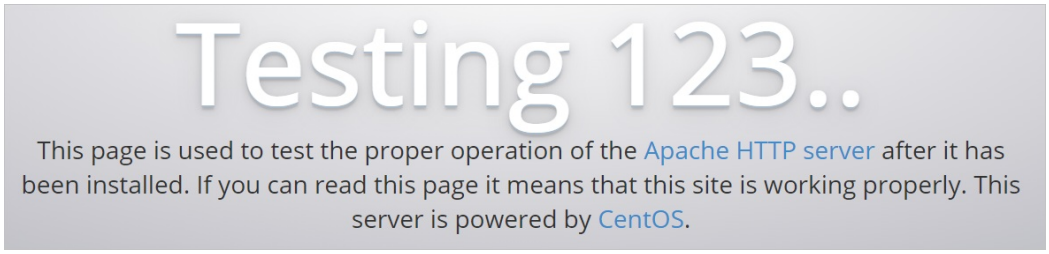
```
Server version: Apache/2.4.6 (CentOS)
Server built:   Apr  2 2020 13:13:23
```

3. Run the following commands in sequence to start Apache and configure Apache to start on system startup:

```
systemctl start httpd
```

```
systemctl enable httpd
```

4. Check the installation result.
   i. Log on to the .
   ii. In the left-side navigation pane, choose .
   iii. On the **Instances** page, find the instance on which you want to build a LAMP environment and copy its public IP address from the **IP Address** column.
   iv. Enter `http://<Public IP address of the ECS instance>` in the address bar of your browser and press the *Enter* key.
   If the following page is displayed, Apache is started.



Step 3: Install and configure MySQL

1. Run the following command to update the YUM repository:

```
rpm -Uvh  http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

2. Run the following command to install MySQL:

> **Note** If you are using an operating system whose kernel version is el8, you may receive the `No match for argument` error message. If this occurs, run the **yum module disable mysql** command to disable the default MySQL module before you install MySQL.

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

3. Run the following command to check the MySQL version:

```
mysql -V
```

   The following example command output indicates that MySQL is installed:

```
mysql  Ver 14.14 Distrib 5.7.31, for Linux (x86_64) using  EditLine wrapper
```

4. Run the following command to start MySQL:

```
systemctl start mysqld
```

5. Run the following commands in sequence to configure MySQL to start on system startup:

```
systemctl enable mysqld
systemctl daemon-reload
```

6. Run the following command to check the initial password of the root account used to log on to

MySQL:

```
grep "password" /var/log/mysqld.log
```

A command output similar to the following one is returned and displays the initial password of the root account. In this example, the initial password of the root account is `+47,uijcojcU`:

```
2020-08-28T03:01:49.848762Z 1 [Note] A temporary password is generated for root@localhost: +47,uijcojcU
```

7. Run the following command to configure the security settings of MySQL:

```
mysql_secure_installation
```

Perform the following operations:

i. Reset the password of the root account.

> **Note** You must keep the password of the root account secure.

```
Enter password for user root: # Enter the initial password of the root account that you obtained in the preceding step.
The existing password for the user account root has expired. Please set a new password.
New password: # Enter a new password that is 8 to 30 characters in length. The password must contain uppercase letters, lowercase letters, digits, and special characters. Supported special characters include ( ) `  ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /
Re-enter new password: # Enter the new password again.
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 # Enter Y and enter the new password again.
New password: # Enter the new password again.
Re-enter new password: # Enter the new password again.
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 # Enter Y to use the new password.
```

ii. Enter *Y* to delete the anonymous user account.

```
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  # Enter Y to delete the anonymous user.
Success.
```

iii. Enter *Y* to deny remote access by the root account.

```
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 # Enter
Y to deny remote access by the root account.
Success.
```

iv. Enter *Y* to delete the test database and the access permissions on the database.

```
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 # Enter Y to delete the test database and the access permissions on the database.
 - Dropping test database...
Success.
 - Removing privileges on test database...
Success.
```

v. Enter *Y* to reload privilege tables.

```
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 # Enter
Y to reload privilege tables.
Success.
All done!
```

Step 4: Install PHP

1. Update the YUM repository.

   i. Run the following commands to add the EPEL repository:

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

   ii. Run the following command to add the Webtatic repository:

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

2. Run the following command to install PHP:

   ```
   yum -y install php70w-devel php70w.x86_64 php70w-cli.x86_64 php70w-common.x86_64 php70w
   -gd.x86_64 php70w-ldap.x86_64 php70w-mbstring.x86_64 php70w-mcrypt.x86_64  php70w-pdo.x
   86_64   php70w-mysqlnd  php70w-fpm php70w-opcache php70w-pecl-redis php70w-pecl-mongodb
   ```

3. Run the following command to check the PHP version:

   ```
   php -v
   ```

   The following example command output indicates that PHP is installed:

```
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
```

4. Run the following command to create a test file in the root directory of the Apache website:

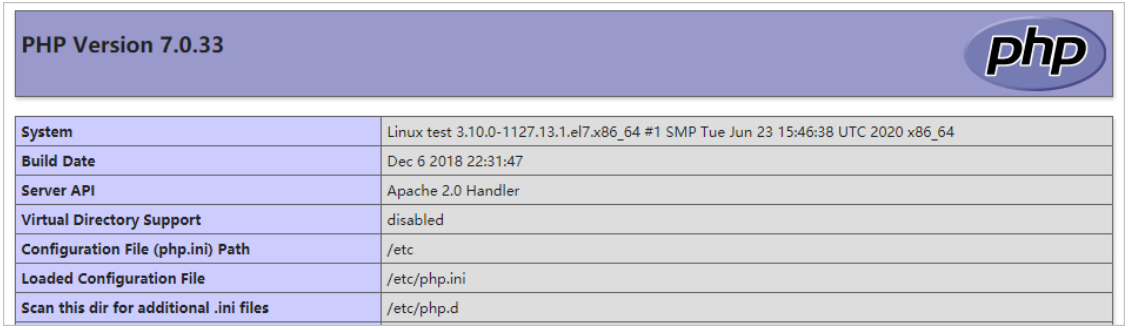
```
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

5. Run the following command to restart Apache:

```
systemctl restart httpd
```

6. Enter `http://<Public IP address of the ECS instance>/phpinfo.php` in the address bar of your browser and press the *Enter* key.

   If the following page is displayed, PHP is installed.



Step 5: Install phpMyAdmin

phpMyAdmin is a MySQL databases management tool that allows you to manage databases conveniently by using web interfaces.

1. Run the following command to prepare a directory to store phpMyAdmin data:

```
mkdir -p /var/www/html/phpmyadmin
```

2. Run the following commands to download and decompress the phpMyAdmin package.
   i. Run the following commands in sequence to switch to the home directory and download the phpMyAdmin package:

```
cd
```

```
wget --no-check-certificate https://files.phpmyadmin.net/phpMyAdmin/4.0.10.20/phpMyAdmin-4.0.10.20-all-languages.zip
```

   ii. Run the following commands in sequence to install the unzip tool and decompress the phpMyAdmin package:

```
yum install -y unzip
```

```
unzip phpMyAdmin-4.0.10.20-all-languages.zip
```

3. Run the following command to copy the phpMyAdmin files to the prepared directory:

```
mv phpMyAdmin-4.0.10.20-all-languages/*  /var/www/html/phpmyadmin
```

4. Enter `http://<Public IP address of the ECS instance>/phpmyadmin` in the address bar of your browser and press the *Enter* key to go to the logon page of phpMyAdmin.
   If the following page is displayed, phpMyAdmin is installed.

5. Enter your MySQL username and password and click **Go**.

3.3. Configure Java Web

3.3.1. Overview of deployment methods

Tomcat is a free, open source Java web server used for web development. It can host Java web applications that consist of Java Servlets, JavaServer pages (dynamic content), HTML pages, JavaScript, style sheets, and pictures (static content).

You can deploy a Java web environment by using one of the following methods:

- Manually deploy the environment

  This method is applicable to users who have basic knowledge of Linux commands. You can manually install and configure a Java web environment. For more information, see Manually deploy a Java web environment on a CentOS 7 instance.

- Use Cloud ToolKit

  Alibaba Cloud Toolkit for Eclipse (Cloud ToolKit) is a free plug-in that can be used within an integrated development environment (IDE). It helps developers deploy applications that are suitable for running on the cloud. After you develop, debug, and test an application on your local PC, you can use this plug-in to deploy the application on an ECS instance.

  For information about how to deploy a Java web environment by using Eclipse, see Use Cloud Toolkit to deploy a Java web environment.

3.3.2. Manually deploy a Java web environment on an instance that runs Alibaba Cloud Linux 2

This topic describes how to manually deploy a Java web environment on an Elastic Compute Service (ECS) instance that runs Alibaba Cloud Linux 2. This topic is applicable to individual users who are new to website construction on ECS instances.

Context

The following instance type and software versions are used in this topic. Operations may vary based on your software versions.

- Instance type: ecs.c6.large
- Operating system: Alibaba Cloud Linux 2.1903 LTS 64-bit
- Java Development Kit (JDK): 1.8.0_292
- Apache Tomcat: 8.5.69

> **Note** In this example, Apache Tomcat 8.5.69 is used. The source code is upgraded on a regular basis. You can manually obtain a version appropriate to your requirements.

Step 1: Make preparations

1. Add inbound rules to the security group of the instance to allow traffic on the required ports.

   For more information, see Add a security group rule. In this example, inbound rules are added to allow traffic on SSH port 22, HTTP port 80, HTTPS port 443, and Apache Tomcat port 8080.

2. Connect to the instance.

   For more information, see Connect to a Linux instance by using a password.

3. Disable the firewall.

   i. Run the **systemctl status firewalld** command to check the state of the firewall.

   ```
   [root@test ~]# systemctl status firewalld
      firewalld.service - firewalld - dynamic firewall daemon
      Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
   eset: enabled)
      Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
        Docs: man:firewalld(1)
    Main PID: 20785 (firewalld)
   ```

   - If the firewall is disabled, it is in the *inactive* state.
   - If the firewall is enabled, it is in the *active* state. In this example, the firewall is in the active state and must be disabled.

ii. Disable the firewall. Skip this step if the firewall is already disabled.

- To temporarily disable the firewall, run the following command:

```
systemctl stop firewalld
```

> **Note** After you run the preceding command, the firewall is disabled. The next time you restart the Linux operating system, the firewall is re-enabled and enters the active state.

- To permanently disable the firewall, run the following commands in turn.

a. To disable the running firewall, run the following command:

```
systemctl stop firewalld
```

b. To stop the firewall service and configure the service not to automatically start on instance startup, run the following command:

```
systemctl disable firewalld
```

> **Note** After you run the preceding commands, the firewall is disabled. The next time you restart the instance, the firewall remains disabled by default. You can re-enable the firewall. For more information, see Firewalld documentation.

4. Disable SELinux.

i. Check the state of SELinux.

```
getenforce
```

Example command output:

```
[root@test ~]# getenforce
Enforcing
```

- If SELinux is disabled, it is in the *Disabled* state.
- If SELinux is enabled, it is in the *Enforcing* state. In this example, SELinux is in the Enforcing state and must be disabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

- To temporarily disable SELinux, run the following command:

```
setenforce 0
```

> **Note** After you run this command, SELinux is disabled. The next time you restart the Linux operating system, SELinux is re-enabled and enters the Enforcing state.

- To permanently disable SELinux, run the following command to open the SELinux configuration file:

```
vi /etc/selinux/config
```

In the */etc/selinux/config* file, move the pointer on the `SELINUX=enforcing` line and press the /key to enter the edit mode. Set `SELINUX` to disabled. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the SELinux configuration file.

> **Note** You can re-enable SELinux. For more information, see Enable or disable SELinux.

iii. Restart the system for the changes to take effect.

5. For security reasons, create a standard user named www to run Apache Tomcat.

```
useradd www
```

6. Create a website root directory.

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

7. Set the owner of the website root directory to www.

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

Step 2: Install JDK 1.8

1. Query the JDK 1.8 package.

```
yum -y list java*
```

2. Install the JDK 1.8 package displayed in the software list.

```
yum -y install java-1.8.0-openjdk-devel.x86_64
```

3. Check the JDK version.

```
java -version
```

JDK version information in the sample command output:

```
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (build 1.8.0_292-b10)
OpenJDK 64-Bit Server VM (build 25.292-b10, mixed mode)
```

4. Configure environment variables.

i. Open the configuration file.

```
vim /etc/profile
```

ii. At the end of the configuration file, press the /key to enter the edit mode.

iii. Add the following information.

> **Note** The `JAVA_HOME` value is the path where JDK is installed. In this example, the **find /usr/lib/jvm -name 'java-1.8.0-openjdk-1.8.0*'** command is run to view the path where JDK is installed.

```
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.292.b10-1.1.al7.x86_64
PATH=$PATH:$JAVA_HOME/bin
CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
export JAVA_HOME CLASSPATH PATH
```

iv. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

v. Apply the environment variables.

```
source /etc/profile
```

Step 3: Install Apache Tomcat

1. Download the Apache Tomcat 8 installation package.

```
wget https://apache.claz.org/tomcat/tomcat-8/v8.5.69/bin/apache-tomcat-8.5.69.tar.gz
```

> **Note** The download address of Apache Tomcat may change. For the download address of the latest version, visit the official Apache Tomcat website.

2. Decompress the Apache Tomcat 8 installation package.

```
tar -zxvf apache-tomcat-8.5.69.tar.gz
```

3. Move the Apache Tomcat installation files to the /user/local/tomcat directory.

```
mv apache-tomcat-8.5.69 /usr/local/tomcat/
```

4. Set the owner of the files to www.

```
chown -R www.www /usr/local/tomcat/
```

The */usr/local/tomcat/* directory contains the following subdirectories:

- *bin*: stores Apache Tomcat script files, such as scripts used to enable and disable Apache Tomcat.
- *conf*: stores various global configuration files of the Apache Tomcat server, among which *server.xml* and *web.xml* are the most important files.
- *webapps*: serves as the main web publishing directory of Apache Tomcat. It stores web application files by default.
- *logs*: stores Apache Tomcat operation log files.

5. Configure the *server.xml* file.
   i. Go to the */usr/local/tomcat/conf/* directory.

      ```
      cd /usr/local/tomcat/conf/
      ```

   ii. Rename the *server.xml* file.

      ```
      mv server.xml server.xml_bk
      ```

   iii. Create a *server.xml* file.
      a. Run the following command to create and open the *server.xml* file:

         ```
         vi server.xml
         ```

      b. Press the /key to add the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/
>
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListene
r"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener
"/>
<Listener className="org.apache.catalina.core.AprLifecycleListener"/>
<GlobalNamingResources>
<Resource name="UserDatabase" auth="Container"
 type="org.apache.catalina.UserDatabase"
 description="User database that can be updated and saved"
 factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
 pathname="conf/tomcat-users.xml"/>
</GlobalNamingResources>
<Service name="Catalina">
<Connector port="8080"
 protocol="HTTP/1.1"
 connectionTimeout="20000"
 redirectPort="8443"
 maxThreads="1000"
 minSpareThreads="20"
 acceptCount="1000"
 maxHttpHeaderSize="65536"
 debug="0"
 disableUploadTimeout="true"
 useBodyEncodingForURI="true"
 enableLookups="false"
 URIEncoding="UTF-8"/>
<Engine name="Catalina" defaultHost="localhost">
<Realm className="org.apache.catalina.realm.LockOutRealm">
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
  resourceName="UserDatabase"/>
</Realm>
<Host name="localhost" appBase="/data/wwwroot/default" unpackWARs="true" autoDe
ploy="true">
<Context path="" docBase="/data/wwwroot/default" debug="0" reloadable="false" c
rossContext="true"/>
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t &amp;quot;%r&amp;quot
; %s %b" />
</Host>
</Engine>
</Service>
</Server>
```

c. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

6. Configure the Java Virtual Machine (JVM) memory parameters.

   i. Run the following command to create and open the */usr/local/tomcat/bin/setenv.sh* file:

```
vi /usr/local/tomcat/bin/setenv.sh
```

ii. Press the /key to add the following content.

Specify the `JAVA_OPTS` parameter to set the JVM memory information and encoding format.

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

iii. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

7. Configure a script to enable Apache Tomcat to run on system startup.

i. Download the script.

> **Note** This script originates from the community and is for reference only. Alibaba Cloud does not make any guarantee, express or implied, with respect to the performance and reliability of the script or the potential impacts of the script operations. If you cannot download the script by running the **wget** command, you can use a browser to access `https://raw.githubusercontent.com/oneinstack/oneinstack/master/init.d/Tomcat-init` to obtain the script content.

```
wget https://raw.githubusercontent.com/oneinstack/oneinstack/master/init.d/Tomcat-i
nit
```

ii. Move and rename Tomcat-init.

```
mv Tomcat-init /etc/init.d/tomcat
```

iii. Grant the execute permissions on the */etc/init.d/tomcat* file.

```
chmod +x /etc/init.d/tomcat
```

iv. Configure the JAVA_HOME script to enable Apache Tomcat to run on system startup.

> **Notice** The JDK version in the script must be the same as the one that was installed. Otherwise, Tomcat cannot start.

```
sed -i 's@^export JAVA_HOME=.*@export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8
.0.292.b10-1.1.al7.x86_64@' /etc/init.d/tomcat
```

8. Run the following commands in sequence to enable Apache Tomcat to run on system startup:

```
chkconfig --add tomcat
```

```
chkconfig tomcat on
```

9. Run the following command to start Apache Tomcat:

```
service tomcat start
```

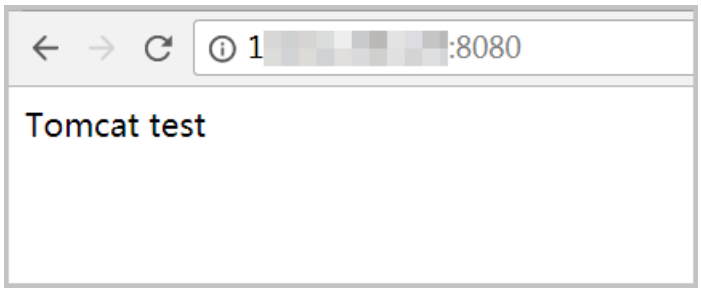
Step 4: Deploy and verify the test project

Upload the WAR package of Java web project files to the website root directory and change the owner of files in the root directory to www. You can use a remote connection tool that has a file transfer feature or build an FTP site to upload project files. In this example, the website root directory is */data/wwwroot/default*. Perform the following operations to create an Apache Tomcat test page in the website root directory and access the page.

1. Deploy the test file.

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

2. Open your browser and enter `http://<Public IP address of the ECS instance>:8080` in the address bar to connect to the ECS instance.
   The page shown in the following figure indicates that Apache Tomcat is installed.

3.3.3. Manually deploy a Java web environment on a CentOS 7 instance

This topic describes how to manually deploy a Java web environment on an Elastic Compute Service (ECS) instance. This topic is applicable to individual users who are new to website construction on ECS instances.

Prerequisites

-
-
-
- To use ECS instances that are located in Chinese mainland regions, make sure that you have completed real-name verification for your account.
- An ECS instance is created. For more information, see Create an instance by using the wizard.

Context

In this topic, the following instance type and software versions are used. Operation may vary based on your instance type and software versions.

- Instance type: ecs.c6.large
- Operating system: CentOS 7.4
- Apache Tomcat: Apache Tomcat 8.5.53

> **Note** In this topic, Apache Tomcat 8.5.53 is used. The source code is constantly upgraded, and you can obtain a version appropriate to your requirements.

- JDK: JDK 1.8.0_241
- FTP tool: WinSCP

Step 1: Download the source code

1. Download Apache Tomcat. For more information, visit Index of /apache/tomcat/tomcat-8/.
2. Download Java Development Kit (JDK).
   i. Download a JDK installation package. For more information, visit Java Downloads.

   > **Note** If you run the **wget** command on an instance to download the JDK installation package and an error is reported when you decompress the package, you can download the JDK installation package to your local computer and upload it to the instance.

   ii.
   iii.
   iv.
   v. Select the region where the instance is deployed.
   vi. On the **Instances** page, find the instance and view its public IP address in the **IP Address** column.
   vii. In WinSCP, connect to the instance by using the public IP address.
   viii. Upload the downloaded Apache Tomcat and JDK installation package to the root directory of the instance.

Step 2: Prepare for installation

1. Add inbound rules to the security group of the instance to allow traffic on the required ports. For more information, see Add a security group rule.

   In this examples, inbound rules are added to the security group to allow traffic on SSH port 22 and HTTP port 8080.
2. Connect to the instance. For more information, see Connect to a Linux instance by using a password.
3. Disable the firewall.
   i. Run the **systemctl status firewalld** command to check the state of the firewall.

```
[root@test ~]# systemctl status firewalld
   firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
eset: enabled)
   Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
     Docs: man:firewalld(1)
 Main PID: 20785 (firewalld)
```

   - If the firewall is disabled, it is in the *inactive* state.
   - If the firewall is enabled, it is in the *active* state. In this example, the firewall is in the active state and must be disabled.

ii. Disable the firewall. Skip this step if the firewall is already disabled.

- To temporarily disable the firewall, run the following command:

```
systemctl stop firewalld
```

> **Note** After you run the preceding command, the firewall is disabled. The next time you restart the Linux operating system, the firewall is re-enabled and enters the active state.

- To permanently disable the firewall, run the following commands in turn.

a. To disable the running firewall, run the following command:

```
systemctl stop firewalld
```

b. To stop the firewall service and configure the service not to automatically start on instance startup, run the following command:

```
systemctl disable firewalld
```

> **Note** After you run the preceding commands, the firewall is disabled. The next time you restart the instance, the firewall remains disabled by default. You can re-enable the firewall. For more information, see Firewalld documentation.

4. Disable SELinux.

i. Check the state of SELinux.

```
getenforce
```

Example command output:

```
[root@test ~]# getenforce
Enforcing
```

- If SELinux is disabled, it is in the *Disabled* state.
- If SELinux is enabled, it is in the *Enforcing* state. In this example, SELinux is in the Enforcing state and must be disabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

- To temporarily disable SELinux, run the following command:

```
setenforce 0
```

> **Note** After you run this command, SELinux is disabled. The next time you restart the Linux operating system, SELinux is re-enabled and enters the Enforcing state.

- To permanently disable SELinux, run the following command to open the SELinux configuration file:

```
vi /etc/selinux/config
```

In the */etc/selinux/config* file, move the pointer on the `SELINUX=enforcing` line and press the /key to enter the edit mode. Set `SELINUX` to disabled. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the SELinux configuration file.

> **Note** You can re-enable SELinux. For more information, see Enable or disable SELinux.

iii. Restart the system for the changes to take effect.

5. To ensure system security, we recommend that you create a standard user to run Apache Tomcat.

   In this example, a standard user named www is created.

```
useradd www
```

6. Run the following command to create a root directory for the Java website:

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

7. Upload the WAR package of Java web project files to the root directory and change the owner of files under the root directory to www.

   In this example, the following commands are run to create a Apache Tomcat test page under the root directory and change the owner of files under the root directory to www:

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

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

Step 3: Install JDK

1. Run the following command to create a directory:

```
mkdir /usr/java
```

2. Run the following commands in sequence to grant the execute permissions on *jdk-8u241-linux-x64.tar.gz* and decompress it to */usr/java*:

```
chmod +x jdk-8u241-linux-x64.tar.gz
```

```
tar xzf jdk-8u241-linux-x64.tar.gz -C /usr/java
```

3. Set environment variables.
   i. Run the `vi /etc/profile` command to open the */etc/profile* file.
   ii. Press the `I` key to add the following content:

```
# set java environment
export JAVA_HOME=/usr/java/jdk1.8.0_241
export CLASSPATH=$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib
export PATH=$JAVA_HOME/bin:$PATH
```

   iii. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.
4. Run the following command to load the environment variables:

```
source /etc/profile
```

5. Run the following command to view the JDK version:

```
java -version
```

   The following command output indicates that JDK is installed.

```
[root@javaweb conf]# java -version
java version "1.8.0_241"
Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
```

Step 4: Install Apache Tomcat

1. Run the following commands in sequence.
   i. Decompress *apache-tomcat-8.5.53.tar.gz*.

```
tar xzf apache-tomcat-8.5.53.tar.gz
```

   ii. Rename the Apache Tomcat directory.

```
mv apache-tomcat-8.5.53 /usr/local/tomcat/
```

   iii. Configure the owner of the file.

```
chown -R www.www /usr/local/tomcat/
```

   The */usr/local/tomcat/* directory contains the following subdirectories:

   - *bin*: stores Apache Tomcat script files, such as scripts used to enable and disable Apache Tomcat.
   - *conf*: stores various global configuration files of the Apache Tomcat server, among which *server.xml* and *web.xml* are the most important files.
   - *webapps*: serves as the main web publishing directory of Apache Tomcat to store web application files by default.
   - *logs*: stores Apache Tomcat operation log files.
2. Configure the *server.xml* file.

i. Run the following command to go to the */usr/local/tomcat/conf/* directory:

```
cd /usr/local/tomcat/conf/
```

ii. Run the following command to rename the *server.xml* file:

```
mv server.xml server.xml_bk
```

iii. Create a *server.xml* file.

a. Run the following command to create and open the *server.xml* file:

```
vi server.xml
```

b. Press the /key to add the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/
>
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListene
r"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener
"/>
<Listener className="org.apache.catalina.core.AprLifecycleListener"/>
<GlobalNamingResources>
<Resource name="UserDatabase" auth="Container"
 type="org.apache.catalina.UserDatabase"
 description="User database that can be updated and saved"
 factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
 pathname="conf/tomcat-users.xml"/>
</GlobalNamingResources>
<Service name="Catalina">
<Connector port="8080"
 protocol="HTTP/1.1"
 connectionTimeout="20000"
 redirectPort="8443"
 maxThreads="1000"
 minSpareThreads="20"
 acceptCount="1000"
 maxHttpHeaderSize="65536"
 debug="0"
 disableUploadTimeout="true"
 useBodyEncodingForURI="true"
 enableLookups="false"
 URIEncoding="UTF-8"/>
<Engine name="Catalina" defaultHost="localhost">
<Realm className="org.apache.catalina.realm.LockOutRealm">
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
  resourceName="UserDatabase"/>
</Realm>
<Host name="localhost" appBase="/data/wwwroot/default" unpackWARs="true" autoDe
ploy="true">
<Context path="" docBase="/data/wwwroot/default" debug="0" reloadable="false" c
rossContext="true"/>
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t &amp;quot;%r&amp;quot
; %s %b" />
</Host>
</Engine>
</Service>
</Server>
```

c. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

3. Configure the Java Virtual Machine (JVM) memory parameters.

   i. Run the following command to create and open the */usr/local/tomcat/bin/setenv.sh* file:

```
vi /usr/local/tomcat/bin/setenv.sh
```

ii. Press the /key to add the following content.

Specify the `JAVA_OPTS` parameter to set the JVM memory information and encoding format.

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

iii. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

4. Configure a script to enable Apache Tomcat to run on system startup.

i. Run the following command to download the script.

> **Note** This script originates from the community and is for reference only. Alibaba Cloud does not make any guarantee, express or implied, with respect to the performance and reliability of the script or the potential impacts of the script operations. If you cannot download the script by running the **wget** command, you can use a browser to access `https://raw.githubusercontent.com/oneinstack/oneinstack/master/init.d/Tomcat-init` to obtain the script content.

```
wget https://raw.githubusercontent.com/oneinstack/oneinstack/master/init.d/Tomcat-i
nit
```

ii. Run the following command to move and rename Tomcat-init:

```
mv Tomcat-init /etc/init.d/tomcat
```

iii. Run the following command to grant the execute permissions on the */etc/init.d/tomcat* file:

```
chmod +x /etc/init.d/tomcat
```

iv. Run the following command to configure the JAVA_HOME script to enable Apache Tomcat to run on system startup.

> **Notice** The JDK version in the script must be the same as that you installed. Otherwise, Apache Tomcat cannot start.

```
sed -i 's@^export JAVA_HOME=.*@export JAVA_HOME=/usr/java/jdk1.8.0_241@' /etc/init.
d/tomcat
```

5. Run the following commands in sequence to enable Apache Tomcat to run on system startup:

```
chkconfig --add tomcat
```

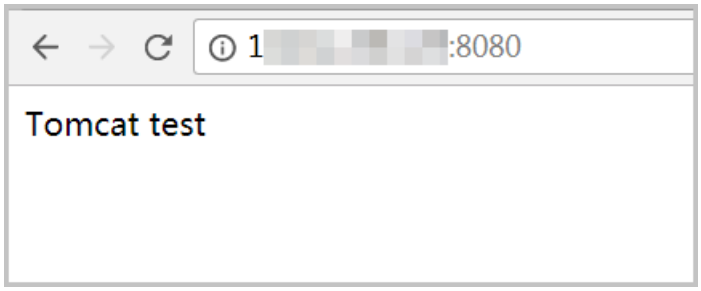
```
chkconfig tomcat on
```

6. Run the following command to start Apache Tomcat:

```
service tomcat start
```

7. Open your browser and enter a URL in the `http://<Public IP address of the ECS instance>:8080` format in the address bar to connect to the instance.

The following response indicates that Apache Tomcat is installed.



What's next

When Apache Tomcat becomes available, we recommend that you configure websites on the instance and map the domain name of the websites to the public IP address of the instance.

3.3.4. Use Cloud Toolkit to deploy a Java web environment

Alibaba Cloud Toolkit for Eclipse (Cloud Toolkit) is a free plug-in used for an integrated development environment (IDE). After you develop, debug, and test an application on your computer, you can use this plug-in to deploy the application to an Elastic Compute Service (ECS) instance.

Prerequisites

- Java Development Kit (JDK) 1.8 or later is downloaded and installed. For more information, visit 64-bit Java for Windows.
- Eclipse IDE 4.5.0 or later is downloaded and installed. For more information, visit the Eclipse website. Eclipse IDE 4.5.0 or later is suitable for Java Enterprise Edition (Java EE) developers.
- An AccessKey pair is created. For more information, see Obtain an AccessKey pair.
- An ECS instance is created. For more information, see Create an instance by using the wizard.

  In these examples, a Linux instance of the ecs.c6.large instance type is used.
- A security group of the Virtual Private Cloud (VPC) type is created. Inbound rules are added to the security group to allow traffic on ports 21, 22, and 80. For more information, see Add a security group rule.
- Windows Secure Copy (WinSCP) or another remote tool that can be used to connect to the Linux operating system is installed.

Context

This topic describes how to install Cloud Toolkit in Eclipse on your Windows computer and then use Cloud Toolkit to deploy a Java application to an ECS instance.

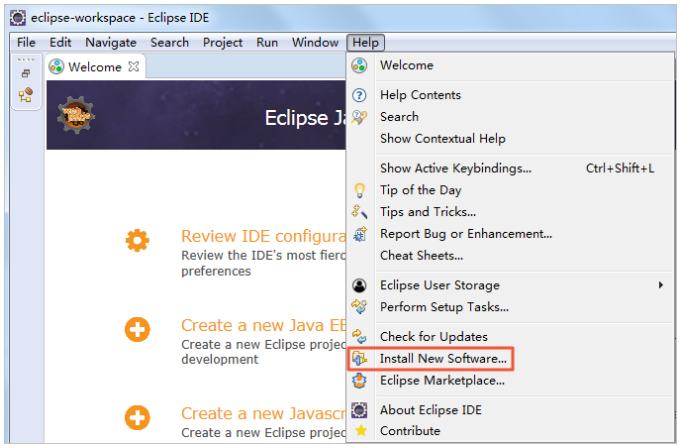
Procedure

1. Step 1: Install Cloud Toolkit on your Windows computer
2. Step 2: Configure an AccessKey pair
3. Step 3: Download and upload a JDK installation package
4. Step 4: Make preparations
5. Step 5: Install JDK

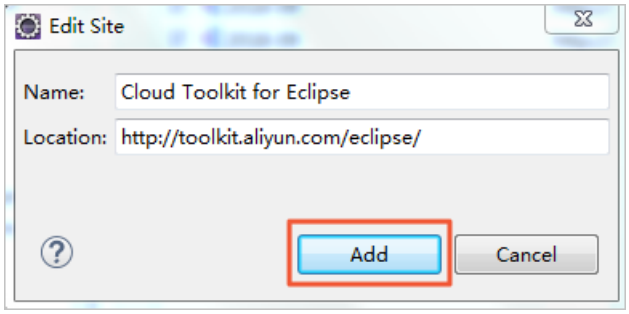
6. Step 6: Install Apache Tomcat
7. Step 7: Deploy a Java application to an ECS instance

Step 1: Install Cloud Toolkit on your Windows computer

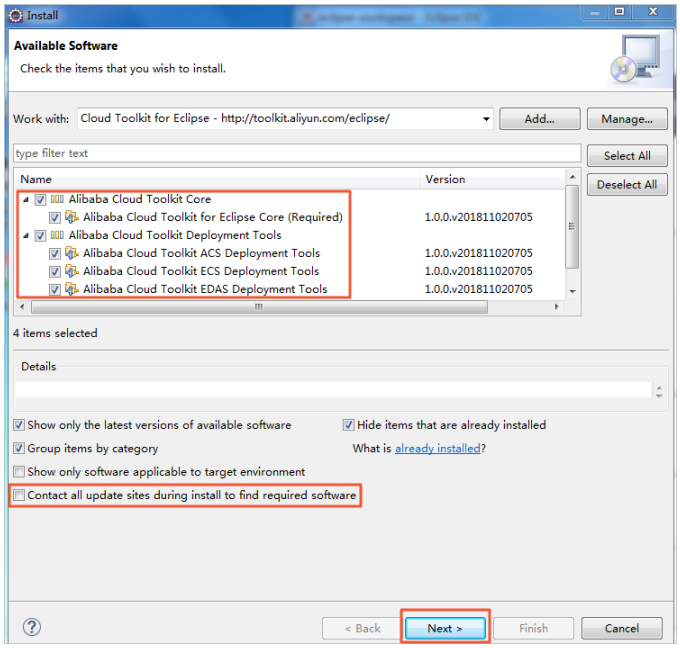
1. Start Eclipse on your Windows computer.
2. In the top navigation bar of the Eclipse window, choose **Help > Install New Software...**



3. Click **Add...**
4. Enter a name such as *Cloud Toolkit for Eclipse* in the Name field, enter *http://toolkit.aliyun.com/eclipse* in the Location field, and then click **Add**.

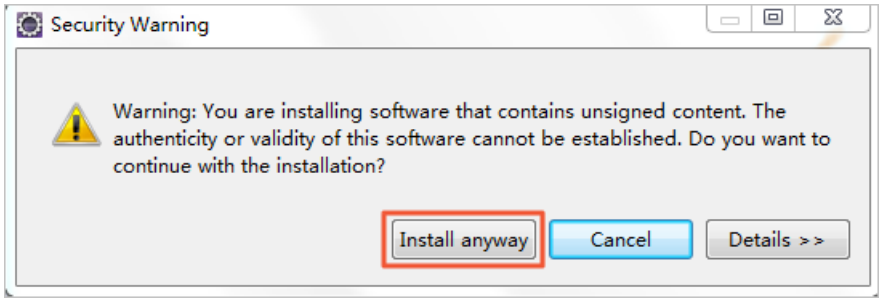


5. In the Name column, select **Alibaba Cloud Toolkit Core** and **Alibaba Cloud Toolkit Deployment Tools**. In the **Details** section, clear **Contact all update sites during install to find required software**. Then, click **Next**.

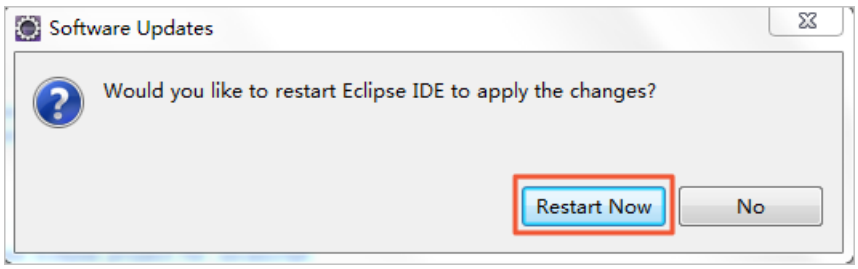


6. Click **Next**.
7. Select **I accept the terms of the license agreement** and click **Finish**.

   You can view the installation progress of Cloud Toolkit in the lower-right corner of the Eclipse window.
8. While Cloud Toolkit is being installed, the **Security Warning** dialog box appears. In the Security Warning dialog box, click **Install anyway**.



9. After Cloud Toolkit is installed, the **Software Updates** dialog box appears. In the Software Updates dialog box, click **Restart Now** to restart Eclipse.



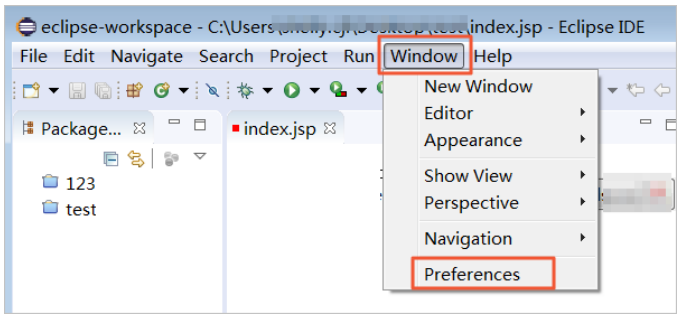
   After specific versions of Eclipse are restarted, they must be activated for use by using an activation code. You must obtain an activation code and activate Eclipse on your own.

Step 2: Configure an AccessKey pair

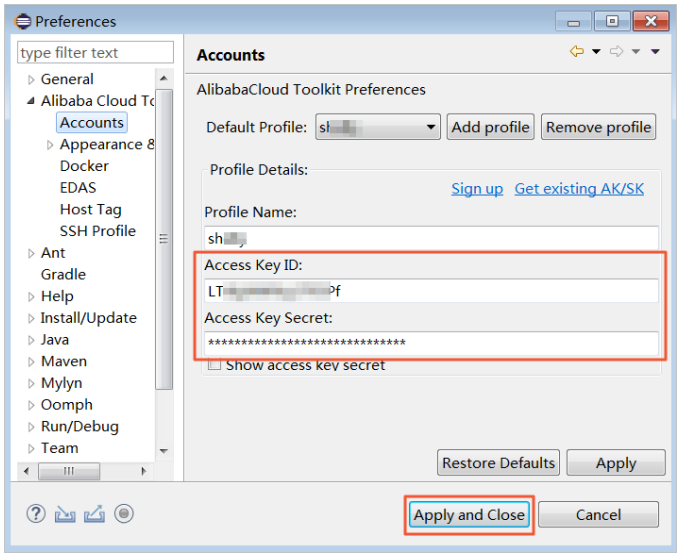
An AccessKey ID identifies a user. An AccessKey secret encrypts the signature string and is the key that the server uses to authenticate the signature string. The AccessKey pair must be kept confidential.

Perform the following operations to configure the AccessKey ID and AccessKey secret.

1. In the top navigation bar of the Eclipse window, choose **Window > Preferences**.



2. In the left-side navigation pane, choose **Alibaba Cloud Toolkit > Accounts**.
3. Specify **AccessKey ID** and **AccessKey Secret**, and click **Apply and Close**.



Step 3: Download and upload a JDK installation package

1. Download Apache Tomcat. For more information, visit Index of /apache/tomcat/tomcat-8/.

> **Note** The source code is constantly upgraded and you can obtain the version that is appropriate to your needs.

2. Download a JDK installation package. For more information, visit Java Downloads.

> **Note** If you download a JDK installation package for Linux to the instance, an error occurs when you decompress the package. To avoid this error, you can download a JDK installation package for Linux to your computer and then upload the package to the instance.

3.
4.
5.
6. On the Instances page, find the Linux instance and view its public IP address in the **IP Address** column.

7. In WinSCP, use the public IP address to connect to the Linux instance. Then, upload the JDK installation package to the root directory of the instance.

Step 4: Make preparations

1. Log on to the Linux instance.

   For more information, see Connect to a Linux instance by using a password.

2. Disable the firewall.

   i. Run the **systemctl status firewalld** command to check the state of the firewall.

```
[root@test ~]# systemctl status firewalld
   firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pr
eset: enabled)
   Active: active (running) since Tue 2018-11-13 10:40:03 CST; 21s ago
     Docs: man:firewalld(1)
 Main PID: 20785 (firewalld)
```

   - If the firewall is disabled, it is in the *inactive* state.
   - If the firewall is enabled, it is in the *active* state. In this example, the firewall is in the active state and must be disabled.

   ii. Disable the firewall. Skip this step if the firewall is already disabled.

   - To temporarily disable the firewall, run the following command:

```
systemctl stop firewalld
```

     Note After you run the preceding command, the firewall is disabled. The next time you restart the Linux operating system, the firewall is re-enabled and enters the active state.

   - To permanently disable the firewall, run the following commands in turn.

     a. To disable the running firewall, run the following command:

```
systemctl stop firewalld
```

     b. To stop the firewall service and configure the service not to automatically start on instance startup, run the following command:

```
systemctl disable firewalld
```

     Note After you run the preceding commands, the firewall is disabled. The next time you restart the instance, the firewall remains disabled by default. You can re-enable the firewall. For more information, see Firewalld documentation.

3. Disable SELinux.

i. Check the state of SELinux.

```
getenforce
```

Example command output:

```
[root@test ~]# getenforce
Enforcing
```

- If SELinux is disabled, it is in the *Disabled* state.
- If SELinux is enabled, it is in the *Enforcing* state. In this example, SELinux is in the Enforcing state and must be disabled.

ii. Disable SELinux. Skip this step if SELinux is already disabled.

- To temporarily disable SELinux, run the following command:

```
setenforce 0
```

> **Note** After you run this command, SELinux is disabled. The next time you restart the Linux operating system, SELinux is re-enabled and enters the Enforcing state.

- To permanently disable SELinux, run the following command to open the SELinux configuration file:

```
vi /etc/selinux/config
```

In the */etc/selinux/config* file, move the pointer on the `SELINUX=enforcing` line and press the /key to enter the edit mode. Set `SELINUX` to disabled. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the SELinux configuration file.

> **Note** You can re-enable SELinux. For more information, see Enable or disable SELinux.

iii. Restart the system for the changes to take effect.

4. Create a user named www to run Tomcat.

```
useradd www
```

5. Create a website root directory.

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

6. Change the owner of the website root directory to www.

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

Step 5: Install JDK

1. Create a directory.

```
mkdir /usr/java
```

2. Decompress the JDK installation package.

In this example, the *jdk-8u241-linux-x64.tar.gz* JDK installation package is decompressed to the */usr*

/java directory.

```
chmod +x jdk-8u241-linux-x64.tar.gz
tar xzf jdk-8u241-linux-x64.tar.gz -C /usr/java
```

3. Set environment variables.
   i. Open the */etc/profile* file.

   ```
   vi /etc/profile
   ```

   ii. Press the `I` key to enter the edit mode.
   iii. Add the following lines to the */etc/profile* file:

   ```
   # set java environment
   export JAVA_HOME=/usr/java/jdk1.8.0_241
   export CLASSPATH=$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib
   export PATH=$JAVA_HOME/bin:$PATH
   ```

   iv. Press `Esc` to exit the edit mode. Enter `:wq` and press the Enter key to save and close the file.
4. Load environment variables.

   ```
   source /etc/profile
   ```

5. View JDK version information.

   ```
   java -version
   ```

   If JDK is installed, JDK version information is displayed.

   ```
   [root@javaweb conf]# java -version
   java version "1.8.0_241"
   Java(TM) SE Runtime Environment (build 1.8.0_241-b07)
   Java HotSpot(TM) 64-Bit Server VM (build 25.241-b07, mixed mode)
   ```

Step 6: Install Apache Tomcat

1. In the system root directory, run the following commands in turn.
   i. Decompress *apache-tomcat-8.5.53.tar.gz*.

   ```
   tar xzf apache-tomcat-8.5.53.tar.gz
   ```

   ii. Move and rename the Tomcat directory.

   ```
   mv apache-tomcat-8.5.53 /usr/local/tomcat/
   ```

   iii. Configure user permissions on the Tomcat directory.

   ```
   chown -R www.www /usr/local/tomcat/
   ```

   The */usr/local/tomcat/* directory contains the following subdirectories:

   - *bin*: stores some Tomcat script files, such as those used to enable and disable the Tomcat service.
   - *conf*: stores various global configuration files of the Tomcat server, among which *server.xml* and

web.xml are the most important files.

- *webapps*: serves as the main web publishing directory of Tomcat. It stores web application files by default.
- *logs*: stores Tomcat operation log files.

2. Configure the *server.xml* file.
   i. Switch to the */usr/local/tomcat/conf/* directory.

   ```
   cd /usr/local/tomcat/conf/
   ```

   ii. Rename the *server.xml* file.

   ```
   mv server.xml server.xml_bk
   ```

   iii. Open the *server.xml* file.

   ```
   vi server.xml
   ```

   iv. Press the /key to enter the edit mode.
   v. Add the following content to the file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/>
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener"/>
<Listener className="org.apache.catalina.core.AprLifecycleListener"/>
<GlobalNamingResources>
<Resource name="UserDatabase" auth="Container"
 type="org.apache.catalina.UserDatabase"
 description="User database that can be updated and saved"
 factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
 pathname="conf/tomcat-users.xml"/>
</GlobalNamingResources>
<Service name="Catalina">
<Connector port="8080"
 protocol="HTTP/1.1"
 connectionTimeout="20000"
 redirectPort="8443"
 maxThreads="1000"
 minSpareThreads="20"
 acceptCount="1000"
 maxHttpHeaderSize="65536"
 debug="0"
 disableUploadTimeout="true"
 useBodyEncodingForURI="true"
 enableLookups="false"
 URIEncoding="UTF-8"/>
<Engine name="Catalina" defaultHost="localhost">
<Realm className="org.apache.catalina.realm.LockOutRealm">
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
  resourceName="UserDatabase"/>
</Realm>
<Host name="localhost" appBase="/data/wwwroot/default" unpackWARs="true" autoDeploy
="true">
<Context path="" docBase="/data/wwwroot/default" debug="0" reloadable="false" cross
Context="true"/>
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t &amp;quot;%r&amp;quot; %s
%b" />
</Host>
</Engine>
</Service>
</Server>
```

vi. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the file.

3. Set Java Virtual Machine (JVM) memory parameters.

   i. Create and open the */usr/local/tomcat/bin/setenv.sh* file.

   ```
   vi /usr/local/tomcat/bin/setenv.sh
   ```

   ii. Press the /key to enter the edit mode.

iii. Add the following content to the file:

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

iv. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the file.

4. Configure a script to run Tomcat on system startup.

i. Download the script.

> **Note** This script originates from the community and is for reference only. Alibaba Cloud does not make any guarantee, express or implied, with respect to the performance and reliability of the script or the potential impacts of the script operations.

```
wget http://raw.githubusercontent.com/oneinstack/oneinstack/master/init.d/Tomcat-init
```

ii. Rename Tomcat-init.

```
mv Tomcat-init /etc/init.d/tomcat
```

iii. Grant the execute permissions on the */etc/init.d/tomcat* file.

```
chmod +x /etc/init.d/tomcat
```

iv. Configure the JAVA_HOME script to run Tomcat on system startup.

> **Notice** The JDK version in the script must be the same as that you installed. Otherwise, Tomcat cannot start.

```
sed -i 's@^export JAVA_HOME=.*@export JAVA_HOME=/usr/java/jdk1.8.0_241@' /etc/init.d/tomcat
```

5. Configure Tomcat to automatically run on system startup.

```
chkconfig --add tomcat
```

```
chkconfig tomcat on
```

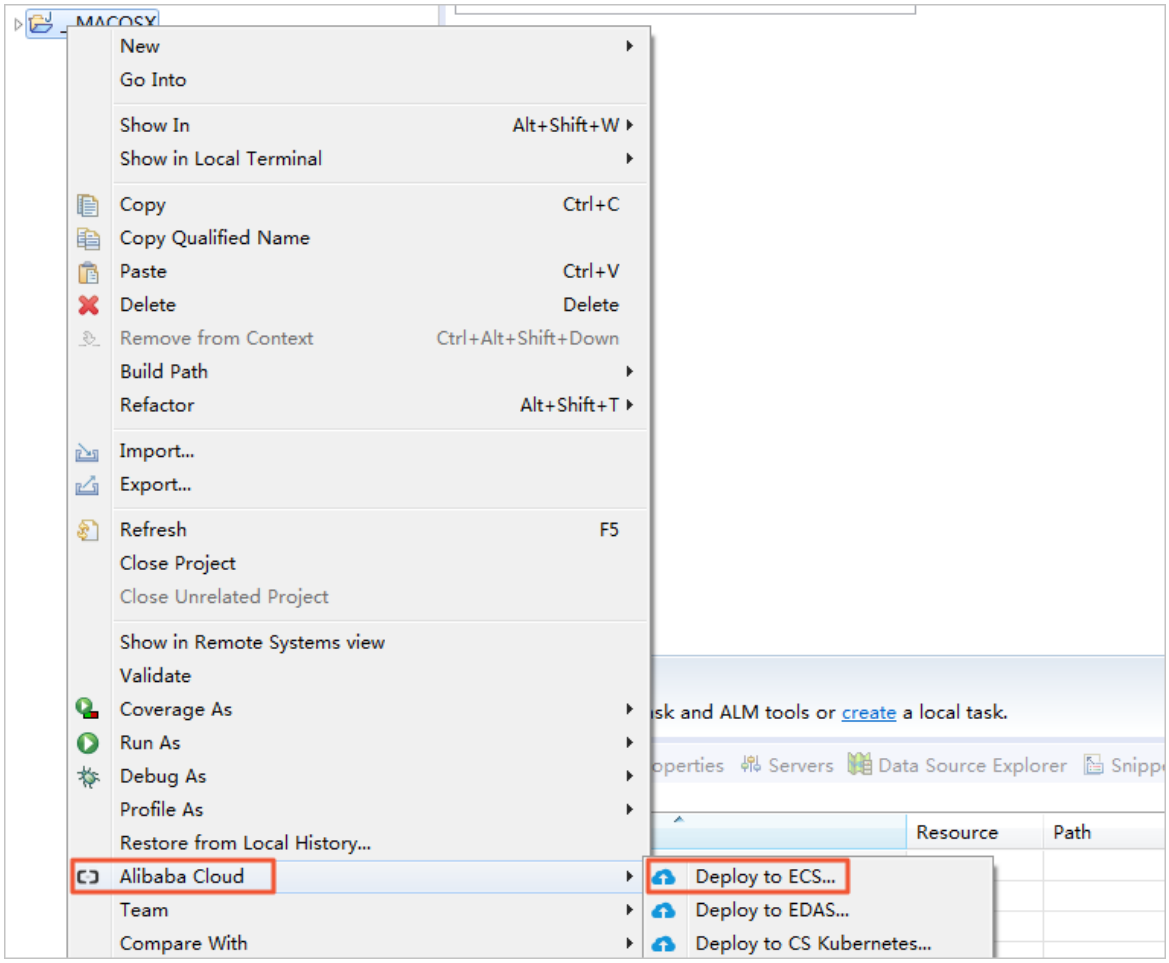
6. Start Tomcat.

```
service tomcat start
```

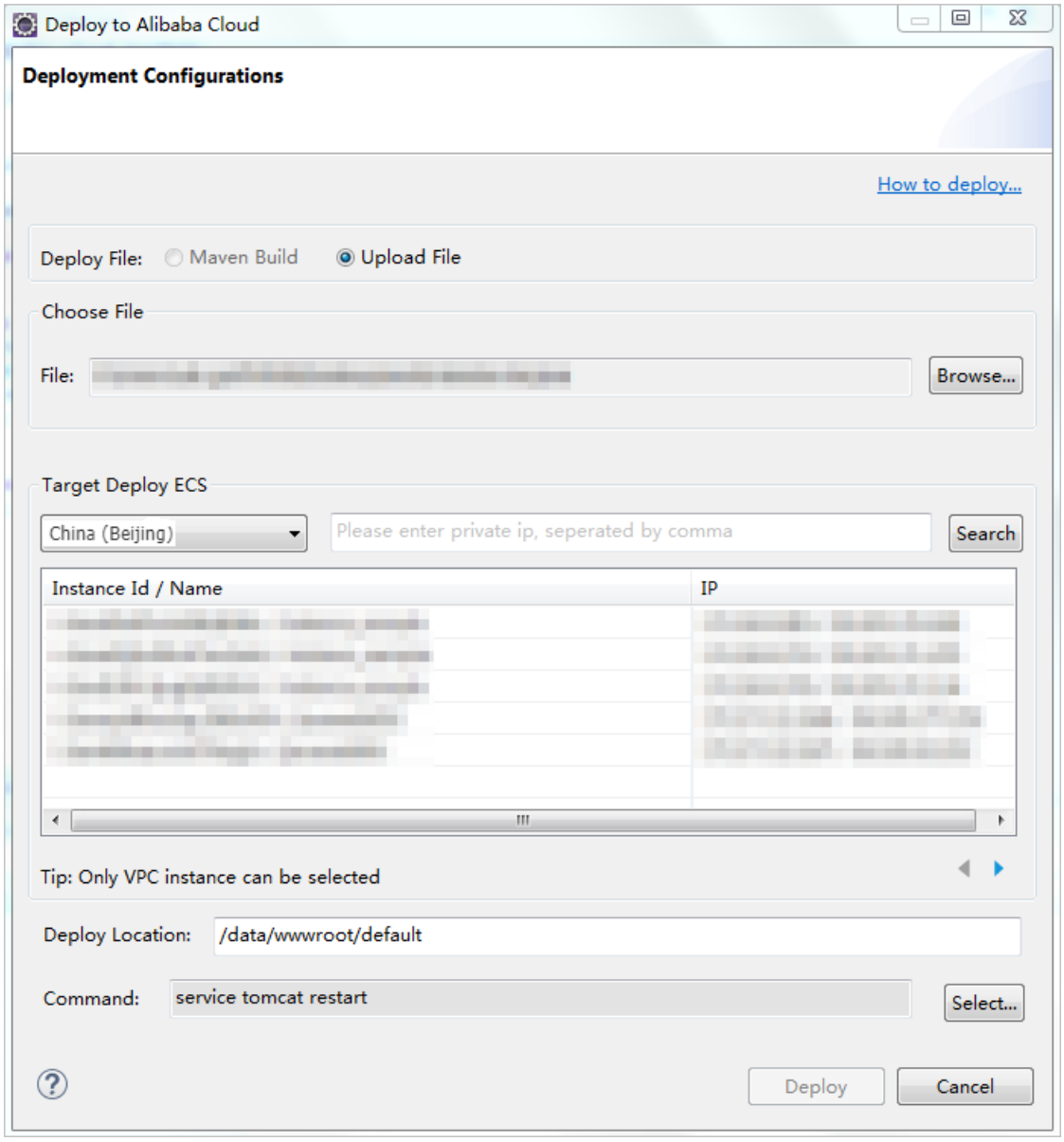
Step 7: Deploy a Java application to an ECS instance

You can use Cloud Toolkit to deploy a Java application to the ECS instance. Then, `Tomcat test` is displayed when you access *http://<Public IP address of the ECS instance>:8080*.

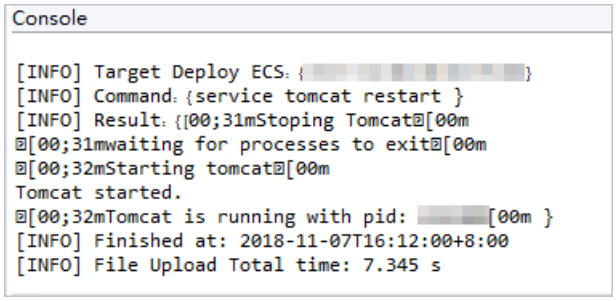
1. In Eclipse, right-click the name of the application project that you want to deploy and choose **Alibaba Cloud > Deploy to ECS...**



2. In the **Deploy to Alibaba Cloud** dialog box, complete the following settings:
   - **Deploy File**: Specify a deployment method. In this example, **Upload File** is selected. If you built the application project by using Maven, select **Maven Build**.
   - **Choose File**: Select the file that you want to deploy.
   - **Target Deploy ECS**: Select a region and an ECS instance.
   - **Deploy Location**: Enter a directory that you deployed on the ECS instance. In this example, */data/wwwroot/default* is used.
   - **Command**: Click **Select...** In the dialog box that appears, and then click **Add...**. Enter a command in the text box. This command is automatically executed after Cloud Toolkit deploys the Java application to the specified directory on the ECS instance. In this example, **service tomcat restart** is entered to restart Tomcat. You can also enter another command based on your business needs.

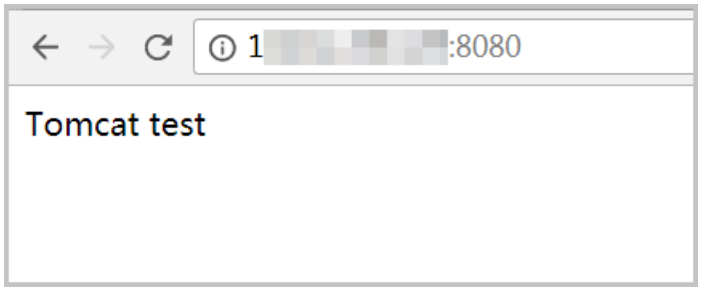


3. Click **Deploy** to start deploying the Java application to the ECS instance.
4. In the **Console** section of Eclipse, you can view the deployment progress.



5. Open your browser and enter `http://<Public IP address of the ECS instance>:8080` in the address bar to connect to the ECS instance.

If the Java application is deployed to the ECS instance by using Cloud Toolkit, the information shown in the following figure is displayed.



What's next

You can modify the Java application in Eclipse, save the code, and then use Cloud Toolkit again to deploy the modified file to the ECS instance.

3.4. Deploy the Node.js environment

3.4.1. Deploy a Node.js environment on an ECS instance that runs Alibaba Cloud Linux 2

This topic describes how to install Node.js and deploy a project on an ECS instance that runs Alibaba Cloud Linux 2.

Prerequisites

Context

Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js is ideal for data-intensive real-time applications that run on distributed devices. The Node.js package manager (npm) is an ecosystem of open source libraries. Node.js is applicable to the following scenarios:

- Real-time applications: instant messaging and real-time push notification applications, such as Socket.IO.
- Distributed applications: applications that perform efficient parallel I/O to consume existing data.
- Utilities: a variety of utilities from frontend compression and deployment applications such as grunt to desktop GUI applications.
- Game applications: real-time and high-concurrency applications in the gaming field, such as the Pomelo framework of NetEase.
- Web rendering applications: applications that use stable interfaces to improve the rendering performance of web pages.
- Consistent frontend and backend programming environments: applications that allow frontend developers to take on server-side development, such as the full-stack JavaScript MongoDB, Express.js, AngularJS, and Node.js. (MEAN) framework.

Step 1: Create and connect to an ECS instance

1. Use the Alibaba Cloud Linux 2.1903 LTS 64-bit public image to create an ECS instance. For more information, see Create an ECS instance.

2. Connect to the ECS instance. For more information, see Connect to a Linux instance by using a password.

Step 2: Deploy a test project

1. Run the following commands in sequence to create a test project file named *example.js*.
   i. Go back to the */root* directory.

   ```
   cd
   ```

   ii. Create the *example.js* test project file.

   ```
   touch example.js
   ```

2. Modify the *example.js* project file.
   i. Run the following command to open the *example.js* file:

   ```
   vim example.js
   ```

   ii. Press the /key to enter the edit mode and add the following content to the *example.js* file.

   In this example, port 3000 is occupied by the project. The command output is `Hello World`. You can configure the project content and a port number based on your business requirements.

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

   iii. After you add the preceding content, press the *Esc* key to exit the edit mode. Enter `:wq` and press the *Enter* key to save and close the file.

3. Run the project and obtain the port number of the project.

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

4. List the ports on which the system is listening.

   ```
   netstat -tpln
   ```

   In this example, port 3000 is included in the command output, which indicates that the project is running normally.

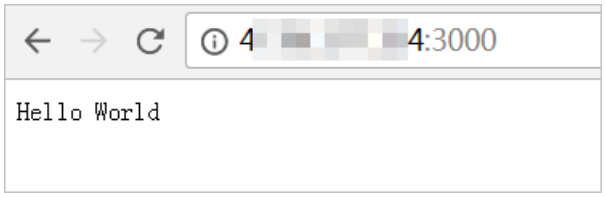
5. Add an inbound rule to the security group of the ECS instance to allow traffic on the specified port.

   In this example, port 3000 is used. For more information about how to add rules to a security group, see Add a security group rule.

6. On your Windows computer or a Windows computer that can access the Internet, open a browser

and enter `http://<Public IP address of the ECS instance>:<Port number>` in the address bar.

In this example, *<Port number>* is 3000. The following page is displayed when you access the project.



3.4.2. Deploy a Node.js environment on a CentOS 7 instance

Node.js is a JavaScript runtime environment built on the Chrome V8 JavaScript engine. You can use Node.js to build online scalable web applications. This topic describes how to install Node.js and deploy a project on an Elastic Compute Service (ECS) instance that runs CentOS 7.8.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

> **Note** In this topic, the CentOS 7.8 public image is used to create an instance.

Context

Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js is ideal for data-intensive real-time applications that run on distributed devices. The Node.js package manager (npm) is an ecosystem of open source libraries. Node.js is applicable to the following scenarios:

- Real-time applications: instant messaging and real-time push notification applications, such as Socket.IO.
- Distributed applications: applications that perform efficient parallel I/O to consume existing data.
- Utilities: a variety of utilities from frontend compression and deployment applications such as grunt to desktop GUI applications.
- Game applications: real-time and high-concurrency applications in the gaming field, such as the Pomelo framework of NetEase.
- Web rendering applications: applications that use stable interfaces to improve the rendering performance of web pages.
- Consistent frontend and backend programming environments: applications that allow frontend developers to take on server-side development, such as the full-stack JavaScript MongoDB, Express.js, AngularJS, and Node.js. (MEAN) framework.

Step 1: Deploy the Node.js environment

1. Connect to the ECS instance.

   For more information, see Connection methodsGuidelines on instance connection.

2. Deploy the Node.js environment.

This topic describes two methods used to deploy the Node.js environment. You can use one of the following methods to deploy the Node.js environment based on your business requirements:

- Method 1: Use NVM to install multiple Node.js versions

  Node Version Manager (NVM) is a software used to manage Node.js versions. You can use NVM to switch among Node.js versions with ease. NVM is suitable for developers who are dedicated to Node.js or users who want to efficiently update or switch among Node.js versions. To install multiple Node.js versions by using NVM, perform the following operations:

  a. Install the distributed version management system Git.

  ```
  yum install git -y
  ```

  b. Use Git to clone the source code of NVM to the local *~/.nvm* directory and check for the latest update.

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

  c. Run the following commands in sequence to configure the environment variables of NVM:

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

  d. Install multiple Node.js versions.

  a. Install v6.9.5.

  ```
  nvm install v6.9.5
  ```

  b. Install v7.4.0.

  ```
  nvm install v7.4.0
  ```

  > **Note** You can run the **nvm list-remote** command to view all versions of Node.js.

  e. Check the installed Node.js versions.

  ```
  nvm ls
  ```

  The following command output indicates that v6.9.5 and v7.4.0 are installed and v7.4.0 is in use:

  ```
           v6.9.5
  ->       v7.4.0
           system
  stable -> 7.4 (-> v7.4.0) (default)
  unstable -> 6.9 (-> v6.9.5) (default)
  ```

  > **Note** You can run the **nvm use <Version number>** command to switch among the Node.js versions. For example, you can run the **nvm use v6.9.5** command to switch to Node.js v6.9.5.

- Method 2: Use binary files to install a Node.js version

The installation package used in this method is a compiled binary file. After you decompress the package, the node and npm files already exist in the *bin* folder. Therefore, you do not need to recompile the binary file. In this example, Node.js v6.9.5 is installed. Perform the following operations:

a. Download the Node.js installation package.

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

b. Decompress the package.

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

c. Run the following commands in sequence to create symbolic links for node and npm.

After you create the symbolic links, you can directly run the node and npm commands in a directory.

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

d. Check the versions of node and npm.

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

Then, the Node.js environment is installed. By default, the software is installed in the */root/node-v6.9.5-linux-x64/* directory.

If you want to install the software to another directory such as */opt/node/*, run the following commands in sequence.

a. Create the */opt/node/* directory.

```
mkdir -p /opt/node/
```

b. Move all files of Node.js to the */opt/node/* directory.

```
mv /root/node-v6.9.5-linux-x64/* /opt/node/
```

c. Remove the symbolic links for node and npm from the source directory.

```
rm -f /usr/local/bin/node
rm -f /usr/local/bin/npm
```

d. Create symbolic links for node and npm in the */opt/node/* directory.

```
ln -s /opt/node/bin/node /usr/local/bin/node
ln -s /opt/node/bin/npm /usr/local/bin/npm
```

Step 2: Deploy a test project

1. Run the following commands in sequence to create a test project file named *example.js*.

i. Go back to the */root* directory.

```
cd
```

ii. Create the *example.js* test project file.

```
touch example.js
```

2. Modify the *example.js* project file.

i. Run the following command to open the *example.js* file:

```
vim example.js
```

ii. Press the *I* key to enter the edit mode and add the following content to the *example.js* file.

In this example, port 3000 is occupied by the project. The command output is `Hello World`. You can configure the project content and a port number based on your business requirements.

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

iii. After you add the preceding content, press the *Esc* key to exit the edit mode. Enter `:wq` and press the *Enter* key to save and close the file.

3. Run the project and obtain the port number of the project.

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

4. List the ports on which the system is listening.

```
netstat -tpln
```

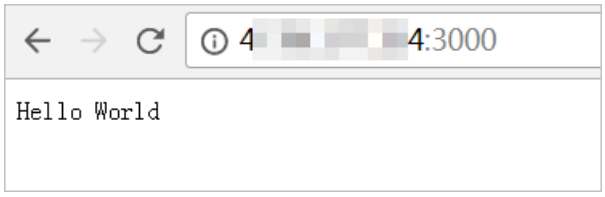
In this example, port 3000 is included in the command output, which indicates that the project is running normally.

5. Add an inbound rule to the security group of the ECS instance to allow traffic on the specified port.

In this example, port 3000 is used. For more information about how to add rules to a security group, see Add a security group rule.

6. On your Windows computer or a Windows computer that can access the Internet, open a browser and enter `http://<Public IP address of the ECS instance>:<Port number>` in the address bar.

In this example, *<Port number>* is 3000. The following page is displayed when you access the project.



3.5. Build a Hadoop environment

This topic describes how to build a Hadoop pseudo-distributed environment on an Elastic Compute Service (ECS) instance that runs a Linux operating system.

Prerequisites

- An ECS Linux instance is created. For more information, see Create an instance by using the wizard.

  In this topic, an ECS instance that has the following configurations is used:

  - Instance type: ecs.g6.large
  - Operating system: CentOS 7.7 64-bit public image
  - Network type: Virtual Private Cloud (VPC)
  - IP address: public IP address

  > **Note** The commands used may vary based on the actual operating system and software versions of your instance. If your software versions or operating system differs from the preceding versions, adjust the commands accordingly.

- The ECS instance is added to security groups that contain rules to allow traffic on ports 8088 and 50070 used by Hadoop. For more information, see Add a security group rule.

Context

Hadoop is an Apache open source distributed framework written in java to efficiently process and store large datasets across clusters. It allows users to develop distributed programs without the need to understand the underlying layer. Hadoop Distributed File System (HDFS) and MapReduce are vital components of Hadoop.

- HDFS is a distributed file system that allows distributed storage and retrieval of application data.
- MapReduce is a distributed computing framework that distributes computing jobs across servers in a Hadoop cluster. Computing jobs are split into map and reduce tasks. JobTracker schedules these tasks for distributed processing.

For more information, visit the Apache Hadoop website.

Procedure

Perform the following steps to build a Hadoop pseudo-distributed environment on the ECS instance:

1. Step 1: Install Java Development Kit (JDK)
2. Step 2: Install Hadoop
3. Step 3: Configure Hadoop
4. Step 4: Configure password-free SSH logon
5. Step 5: Start Hadoop

Step 1: Install Java Development Kit (JDK)

1. Connect to the ECS instance.

   For more information, see Connection methodsGuidelines on instance connection.

2. Run the following command to download the JDK 1.8 installation package:

```
wget https://download.java.net/openjdk/jdk8u41/ri/openjdk-8u41-b04-linux-x64-14_jan_2020.tar.gz
```

3. Run the following command to decompress the downloaded installation package:

```
tar -zxvf openjdk-8u41-b04-linux-x64-14_jan_2020.tar.gz
```

4. Run the following command to move and rename the folder to which the JDK 1.8 installation files are extracted.

   In this example, the folder is renamed `java8`. You can specify a different name for the folder based on your business requirements.

```
mv java-se-8u41-ri/ /usr/java8
```

5. Run the following commands to configure Java environment variables.

   If your specified name of the folder to which the JDK 1.8 installation files are extracted is not java8, replace `java8` in the following commands with the actual folder name:

```
echo 'export JAVA_HOME=/usr/java8' >> /etc/profile
echo 'export PATH=$PATH:$JAVA_HOME/bin' >> /etc/profile
source /etc/profile
```

6. Run the following command to check whether JDK is installed:

```
java -version
```

   A command output similar to the following one indicates that JDK 1.8 is installed:

```
openjdk version "1.8.0_41"
OpenJDK Runtime Environment (build 1.8.0_41-b04)
OpenJDK 64-Bit Server VM (build 25.40-b25, mixed mode)
```

Step 2: Install Hadoop

1. Run the following command to download the Hadoop installation package:

```
wget https://mirrors.bfsu.edu.cn/apache/hadoop/common/hadoop-2.10.1/hadoop-2.10.1.tar.gz
```

2. Run the following command to decompress the Hadoop installation package to the */opt/hadoop* path:

```
tar -zxvf hadoop-2.10.1.tar.gz -C /opt/
mv /opt/hadoop-2.10.1 /opt/hadoop
```

3. Run the following commands to configure Hadoop environment variables:

```
echo 'export HADOOP_HOME=/opt/hadoop/' >> /etc/profile
echo 'export PATH=$PATH:$HADOOP_HOME/bin' >> /etc/profile
echo 'export PATH=$PATH:$HADOOP_HOME/sbin' >> /etc/profile
source /etc/profile
```

4. Run the following commands to modify the *yarn-env.sh* and *hadoop-env.sh* configuration files:

```
echo "export JAVA_HOME=/usr/java8" >> /opt/hadoop/etc/hadoop/yarn-env.sh
echo "export JAVA_HOME=/usr/java8" >> /opt/hadoop/etc/hadoop/hadoop-env.sh
```

5. Run the following command to check whether Hadoop is installed:

```
hadoop version
```

A command output similar to the following one indicates that Hadoop is installed:

```
Hadoop 2.10.1
Subversion https://github.com/apache/hadoop -r 1827467c9a56f133025f28557bfc2c562d78e816
Compiled by centos on 2020-09-14T13:17Z
Compiled with protoc 2.5.0
From source with checksum 3114edef868f1f3824e7d0f68be03650
This command was run using /opt/hadoop/share/hadoop/common/hadoop-common-2.10.1.jar
```

Step 3: Configure Hadoop

1. Modify the *core-site.xml* configuration file of Hadoop.
   i. Run the following command to open the core-site.xml file:

   ```
   vim /opt/hadoop/etc/hadoop/core-site.xml
   ```

   ii. Press the `I` key to enter the edit mode.
   iii. In the `configuration` section, add the following content:

   ```
   <property>
       <name>hadoop.tmp.dir</name>
       <value>file:/opt/hadoop/tmp</value>
       <description>location to store temporary files</description>
   </property>
   <property>
       <name>fs.defaultFS</name>
       <value>hdfs://localhost:9000</value>
   </property>
   ```

   iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.
2. Modify the *hdfs-site.xml* configuration file of Hadoop.
   i. Run the following command to open the hdfs-site.xml file:

   ```
   vim /opt/hadoop/etc/hadoop/hdfs-site.xml
   ```

   ii. Press the `I` key to enter the edit mode.

iii. In the `configuration` section, add the following content:

```
<property>
    <name>dfs.replication</name>
    <value>1</value>
</property>
<property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/opt/hadoop/tmp/dfs/name</value>
</property>
<property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/opt/hadoop/tmp/dfs/data</value>
</property>
```

iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.

Step 4: Configure password-free SSH logon

1. Run the following command to create a public key and a private key:

```
ssh-keygen -t rsa
```

A command output similar to the following one indicates that the public and private keys are created:

```
[root@iZbp1chrrv37a2kts7sydsZ ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:gjWO5mgARst+O5VUaTnGs+LxVhfmCJnQwKfEBTro2oQ root@iZbp1chrrv37a2kts7s****
The key's randomart image is:
+---[RSA 2048]----+
| .  o+Bo=        |
|o o .+.#   o     |
|.= o..B = + .    |
|=.  oO.o o o     |
|Eo..=o* S .      |
|.+.+o. +         |
|. +o. .          |
| .  .            |
|                 |
+----[SHA256]-----+
```

2. Run the following command to add the public key to the *authorized_keys* file:

```
cd .ssh
cat id_rsa.pub >> authorized_keys
```

Step 5: Start Hadoop

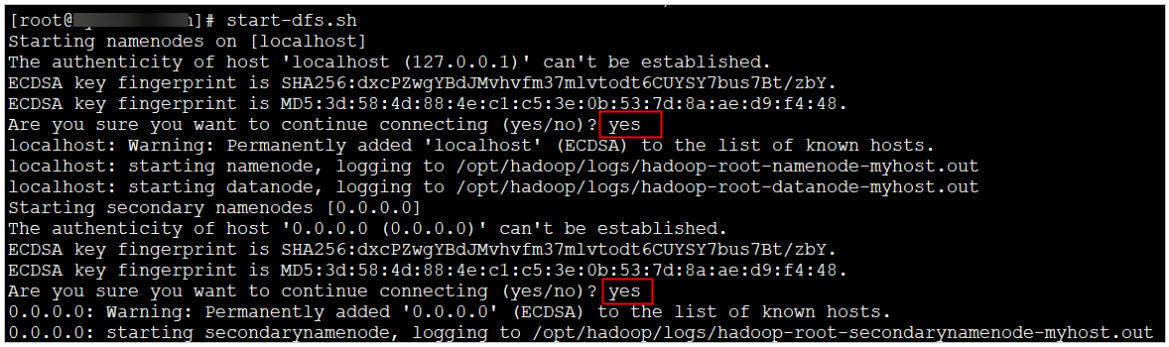
1. Run the following command to initialize `namenode`:

```
hadoop namenode -format
```

2. Run the following commands in sequence to start Hadoop:

```
start-dfs.sh
```

At the prompts that appear, enter `yes`, as shown in the following figure.




```
start-yarn.sh
```

A command output similar to the following one is returned:

```
[root@iZbp1chrrv37a2kts7s**** .ssh]# start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /opt/hadoop/logs/yarn-root-resourcemanager-iZbp1ch
rrv37a2kts7sydsZ.out
localhost: starting nodemanager, logging to /opt/hadoop/logs/yarn-root-nodemanager-iZbp
1chrrv37a2kts7sydsZ.out
```

3. Run the following command to view the processes that are started:

```
jps
```

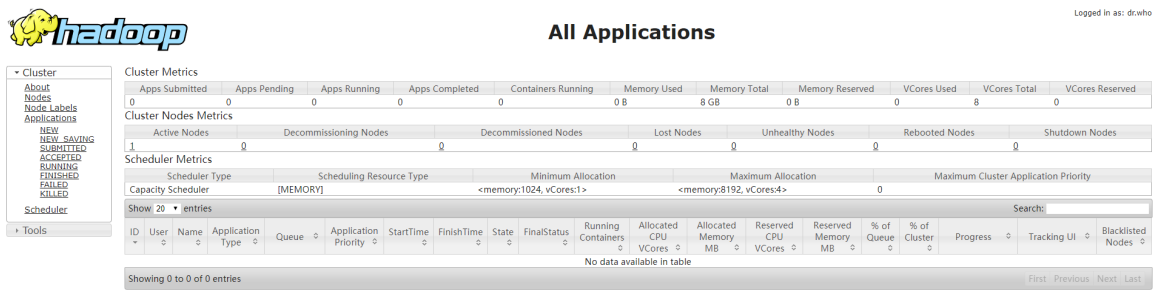
The following processes are started:

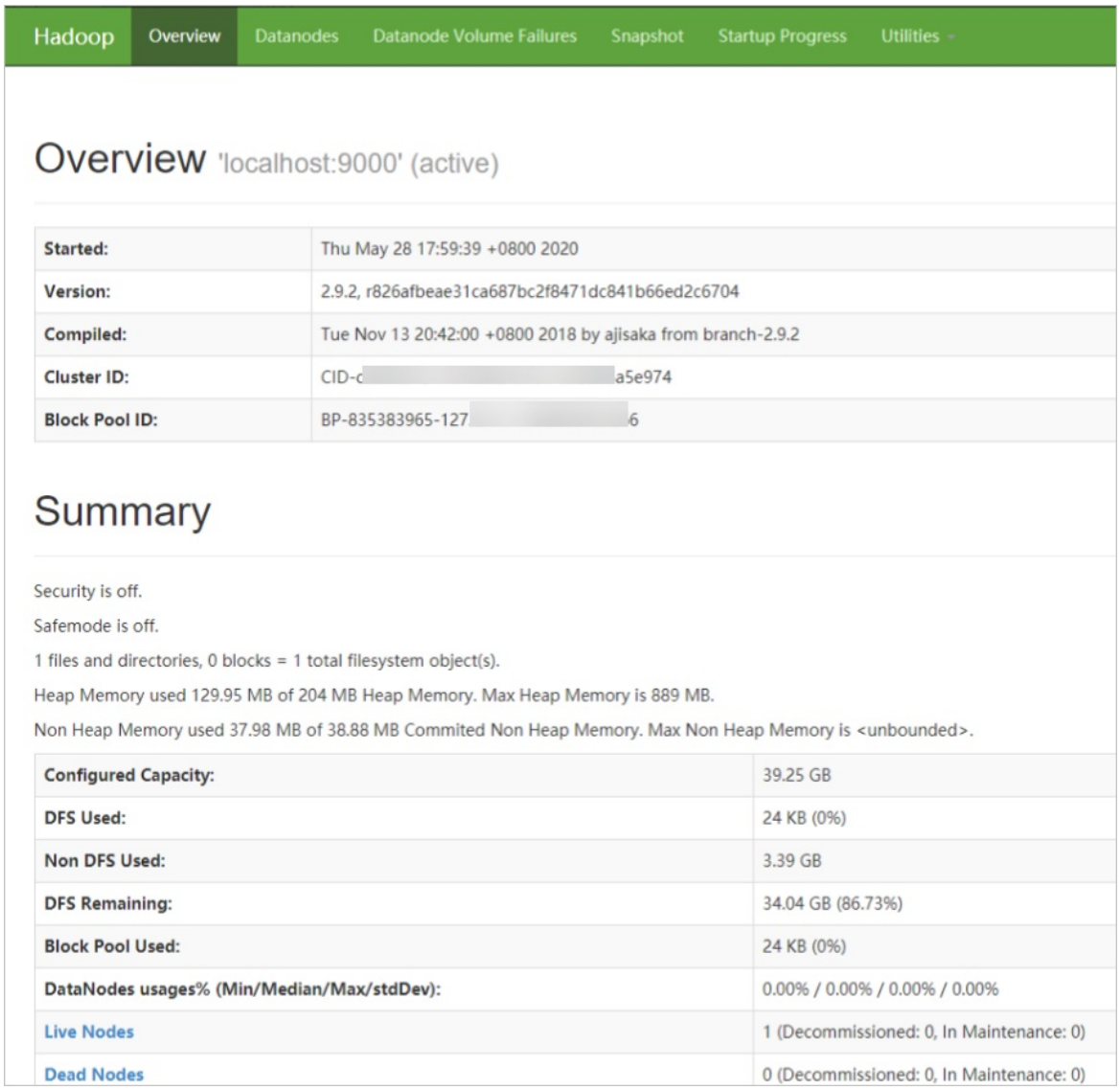
```
[root@iZbp1chrrv37a2kts7s**** .ssh]# jps
11620 DataNode
11493 NameNode
11782 SecondaryNameNode
11942 ResourceManager
12344 Jps
12047 NodeManager
```

4. Use a browser to access `http://<Public IP address of the ECS instance>:8088` and `http://<Public IP address of the ECS instance>:50070`.

If the Hadoop pseudo-distributed environment is built, the page shown in the following figure is displayed.

> **Notice** Make sure that security group rules of the ECS instance allow inbound traffic to ports 8088 and 50070 used by Hadoop. Otherwise, the Hadoop pseudo-distributed environment cannot be accessed. For more information, see Add a security group rule.





3.6. Deploy the Windows application environment by replacing images

You can deploy the Windows application environment on an Elastic Compute Service (ECS) instance by using an image that contains the operating system and applications. This topic describes how to deploy the Windows application environment on an ECS instance by replacing the image of the instance.

Prerequisites

The instance is stopped. If the instance is not stopped, stop it first. For more information, see Stop an instance.

> Notice

Context

Alibaba Cloud Marketplace images contain operating systems and applications and can be used to deploy application environments on ECS instances. For more information, see Alibaba Cloud Marketplace.

You can use one of the following methods to use Alibaba Cloud Marketplace images that contain the Windows application environment:

- If you are creating an instance, you can select an Alibaba Cloud Marketplace image that contains the Windows application environment. For more information, see Creation method overview.
- If you are using an existing instance, you can replace the operating system of the instance by using an Alibaba Cloud Marketplace image that contains the Windows application environment.

Take note of the following items:

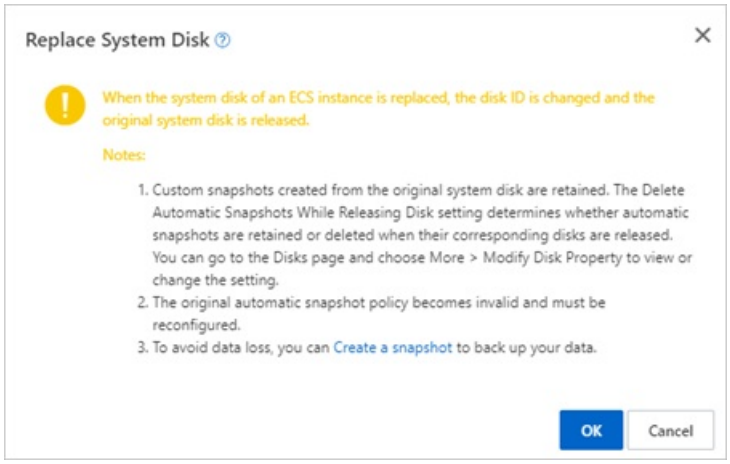
- The procedure described in this topic applies to existing ECS instances. You can deploy the Windows application environment on an existing instance by changing its image to an Alibaba Cloud Marketplace image. This tutorial describes only the general procedure for using Alibaba Cloud Marketplace images. The buy page of Alibaba Cloud Marketplace images contains the how-to guide. We recommend that you read the guide before you use the images.
- You cannot install or deploy virtualization software such as Kernel-based Virtual Machine (KVM), Xen, or VMware on ECS instances.
- After the operating system of an instance is replaced, the original system disk is released and its data is lost and cannot be restored. Exercise caution when you replace the operating system. For more information, see Replace the operating system of an instance by using a non-public image.

  > **Note** When the operating system of an instance is replaced, the data on its data disks is not affected.

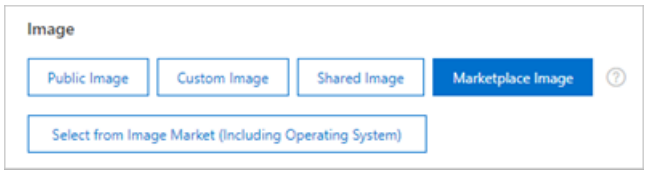
Procedure

1.
2.
3.
4. Replace the system disk of the ECS instance.
   i. Find the instance for which you want to replace the system disk. In the **Actions** column, choose **More > Disk and Image > Replace System Disk**.

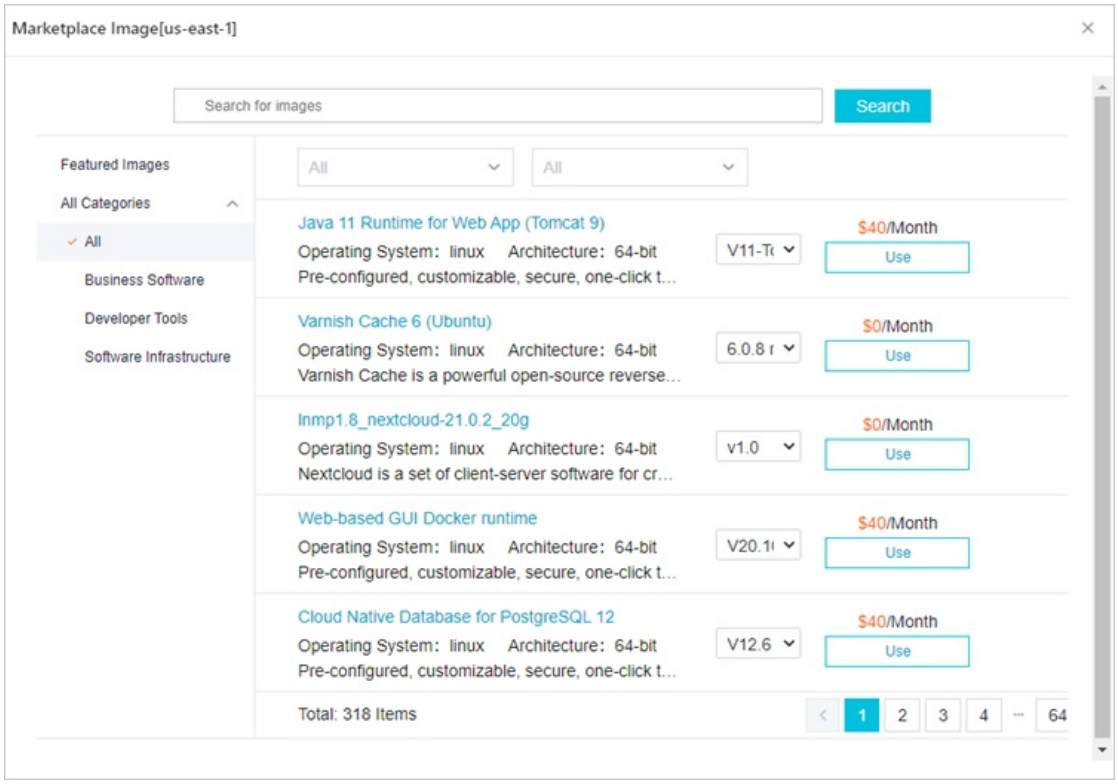
ii. In the **Replace System Disk** message, click **OK**.



iii. On the **Change Operating System** page, set **Image** to **Marketplace Image** and click **Select from Image Market (Including Operating System)**.

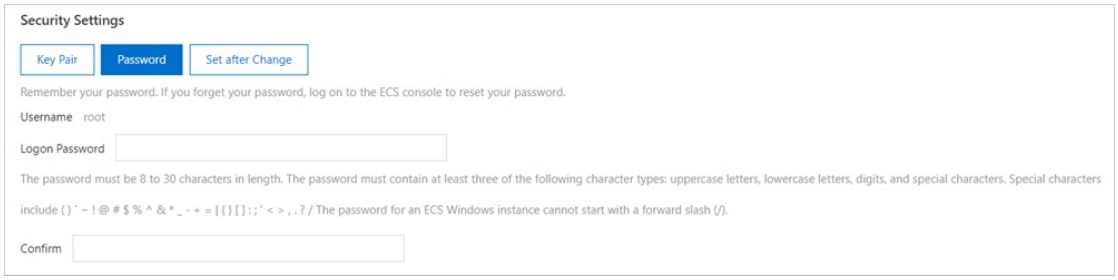


iv. In the left-side navigation pane of the Marketplace Image dialog box, select All Categories or enter the image name that you want to use and click **Search**. Select the image and click **Use**.



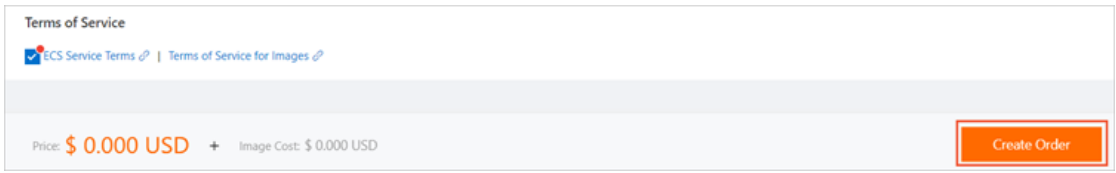
v. In the **System Disk** section, modify the size of the system disk or use the default setting.

vi. In the **Security Settings** section, click **Password**, enter your password in the **Logon Password** field, and then enter the password again in the **Confirm** field.



vii. In the **Terms of Service** section, read and select *ECS Service Terms* and *Terms of Service for Images*.

viii. In the lower-right corner of the **Change Operating System** page, click **Create Oder**.



Result

After you complete the payment, the `Congratulations, purchase successfully` message appears, which indicates that the operating system of the instance is replaced.

The instance automatically starts after the operating system is replaced. You can go back to the **Instances** page, find and connect to the instance, and use the new Windows application environment. For more information about how to connect to an instance, see Connection methodsGuidelines on instance connection.

4.Build a website

4.1. Build a WordPress blog platform

4.1.1. Create a WordPress environment by using ROS

Alibaba Cloud Resource Orchestration Service (ROS) allows you to use templates to create a group of Alibaba Cloud resources. The ROS template is a JSON file used to specify the resources that you want to create. This topic describes how to use ROS templates to create a WordPress environment based on Elastic Compute Service (ECS) and ApsaraDB RDS.

Prerequisites

-
- The first time you use ROS, you are prompted to activate it. ROS is a free service. You can activate ROS free of charge.

Context

ApsaraDB RDS is a stable, reliable, and scalable online database service provided by Alibaba Cloud. ApsaraDB RDS supports database engines such as MySQL, SQL Server, and PostgreSQL. It provides a complete set of solutions for scenarios such as disaster recovery, backup, restoration, monitoring, and migration to reduce your O&M burdens. For more information, see What is ApsaraDB RDS?

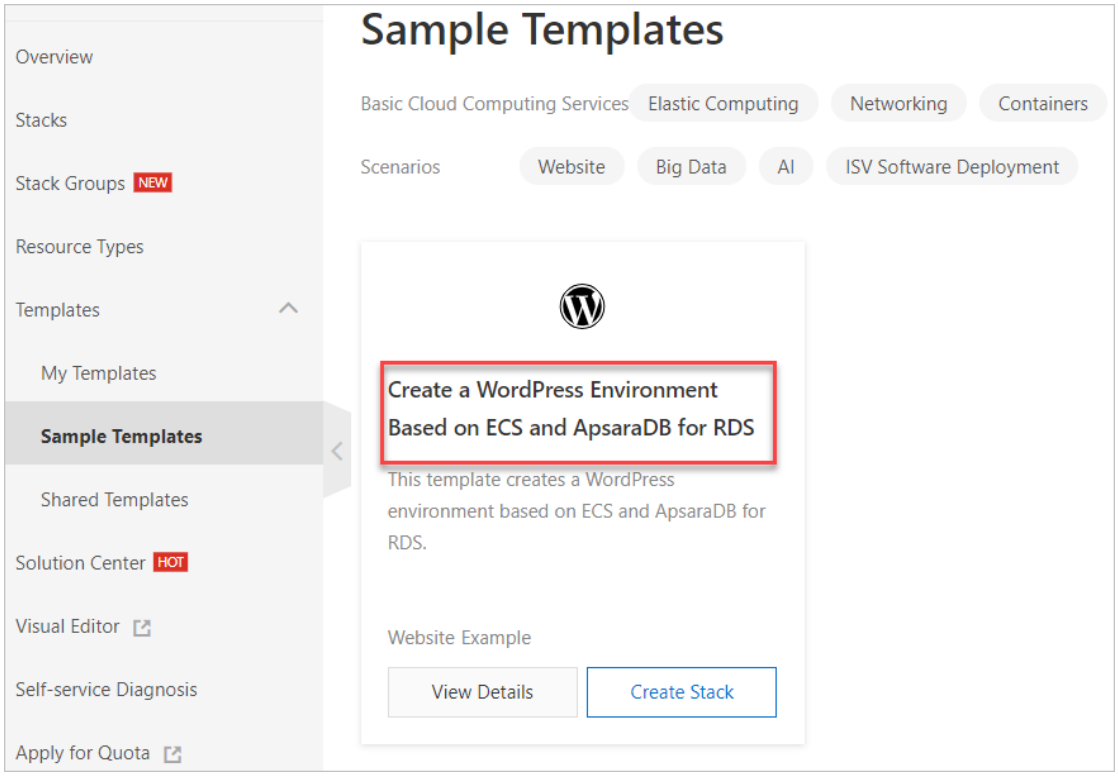
This topic describes how to create a WordPress environment by using the **Create a WordPress Environment Based on ECS and ApsaraDB for RDS** template.

Procedure

1. Log on to the ROS console.
2. Select the template that you want to use.
   i. In the left-side navigation pane, choose **Templates > Sample Templates**.

      The common templates provided by ROS are displayed on the **Sample Templates** page.

ii. Find the **Create a WordPress Environment Based on ECS and ApsaraDB for RDS** template.



iii. (Optional) Click **View Details** to view the template in the JSON format.

The following table describes the top-level fields in the JSON file.

Top-level field	Description
`"ROSTemplateFormatVersion" : "2015-09-01"`	Specifies the version of the template.
`"Parameters" : { }`	Specifies some parameters of the template. In this example, this field specifies the default image ID and instance type.
`"Resources" : { }`	Specifies the Alibaba Cloud resources that you can use the template to create. In this example, this field declares that the resources to be created include an ECS instance and a security group. The properties of these resources are defined in the `Parameters` field.
`"Outputs": { }`	Specifies the resource information that the stack generates after the specified resources are created. In this example, the stack generates the ECS instance ID, public IP address, and security group ID.

> **Note** For more information about the sample templates of ROS, see Template structure.

3. Click **Create Stack**.
4. Configure parameters.
   i. In the top navigation bar, select a region.
   ii. Configure the parameters in the stack template.

The following table describes the stack template parameters.

Parameter		Description
Stack Name		The name of the stack. The stack name must be unique and cannot be modified after the stack is created.
VPC	**VPC CIDR Block**	The private CIDR block of the virtual private cloud (VPC). For more information, see Plan networks.
	VSwitch Availability Zone	The ID of the zone in which to create the resource.

Parameter		Description
	VSwitch CIDR Block	The CIDR block of the vSwitch. The CIDR block of the vSwitch must fall within the CIDR block of the VPC to which the vSwitch belongs and cannot overlap with the CIDR block of an existing vSwitch. For more information, see Plan networks.
ECS	**Instance Type**	The instance type of the ECS instance. For more information about ECS instance types, see Instance family.
	Image	The ID of the image that ROS uses to create the ECS instance.
	Instance Password	The logon password of the ECS instance. The password must be 8 to 30 characters in length and must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. Special characters include `( ) ` ~ ! /@#$%^&*-_+=|{}[]:;'<>,.?` . **Note** Passwords for Windows instances cannot start with a forward slash (/).
RDS	**DB Instance Class**	The type of the RDS instance.
	Engine	The database engine that you want to use.
	DB Instance Storage	The storage capacity of the RDS instance.
	DB Name	The name of the WordPress database.
	DB Username	The username of the WordPress database.
	DB Password	The password used to access the WordPress database. The password must be 6 to 32 characters in length and can contain letters, digits, and underscores (_).

iii. Click **Next**.

iv. Configure stack parameters.

The following table describes the stack parameters.

Parameter	Description
Stack Policy (Optional)	The stack policy.
Rollback on Failure	Specifies whether to roll back the stack if the stack cannot be created. ■ If you select **Enabled**, ROS deletes the created resources when errors such as creation timeout occur during the creation process. ■ If you select **Disabled**, ROS does not delete the created resources when errors such as creation timeout occur during the creation process.
Timeout Period	Specifies the timeout period of the stack creation request. If the specified resources cannot be created within the period, the creation operation times out.

v. Click **Next**.

vi. Check whether the stack parameters are correctly configured.

5. Click **Create**.

Result

In the left-side navigation pane, click **Stacks**. In the top navigation bar, select the region of your created stack from the drop-down list to view stack information such as the state. If the state of the stack is **Created**, the stack is created.

What's next

Click the stack name to go to the stack details page. Click the following tabs to view information about the stack.

- **Stack Information**: The basic information of the stack, such as the state and timeout period is displayed.
- **Events**: The records of the operations that ROS performs during stack creation are displayed. The causes of failed operations are also displayed in the list.
- **Resources**: All resources of the stack are displayed.
- **Template**: The original template of the stack is displayed.

4.1.2. Manually build a WordPress website on a Windows ECS instance

This topic describes how to build a WordPress website on an ECS instance that is running a Windows operating system.

Prerequisites

-
- A security group of the VPC type is created. Inbound rules are added for the security group to allow traffic on ports 80 and 3389. For more information about how to add security group rules, see Add a security group rule.
- A Windows ECS instance is created and deployed with the web environment. The following software versions are used in this tutorial:
  - Operating system: Windows Server 2012 R2 64-bit
  - Internet Information Services (IIS): 8.5
  - PHP: 7.0.28
  - MySQL: 5.5
  - WordPress: 5.3.2

> **Note** If you use software versions different from the preceding versions, you may need to adjust parameter settings.

Build a WordPress website

1. Use the ECS console to connect to the ECS instance that is deployed with the web environment and download the WordPress installation package.
   i. Connect to the ECS instance.
   ii. Download the WordPress installation package from the official WordPress website.

   Version 5.3.2 is used in this tutorial.

   > **Note** If you download WordPress on an ECS instance that is located in a mainland China region and the `429 Too Many Requests` error is reported, we recommend that you try multiple times or download the WordPress installation package from a third-party website.

   iii. Decompress the WordPress installation package.

   In this tutorial, the WordPress installation package is decompressed to `C:\wordpress`.
2. Create a MySQL database for the WordPress website that you want to build.
   i. Go to the bin folder in the MySQL installation directory, right-click a blank area in this folder when you press and hold the `Shift` key, and then select **Open command window here**.
   ii. Log on to the MySQL database.

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

   iii. Create a database for the WordPress website.

   In this tutorial, the database that is created for the WordPress website is named wordpress.

   ```
   create database wordpress;
   ```
3. Configure the WordPress website.
   i. In the `C:\wordpress` directory, find the `wp-config-sample.php` file, copy it, and name the file copy `wp-config.php`.

ii. Use the text editor to open the `wp-config.php` file and modify information related to the wordpress database.

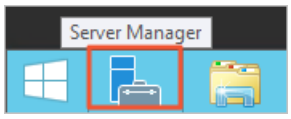
The following figure shows an example.

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** MySQL database username */
define( 'DB_USER', 'root' );

/** MySQL database password */
define( 'DB_PASSWORD', 'password123' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );
```

iii. Save the `wp-config.php` file.

4. Add the WordPress website to Server Manager.

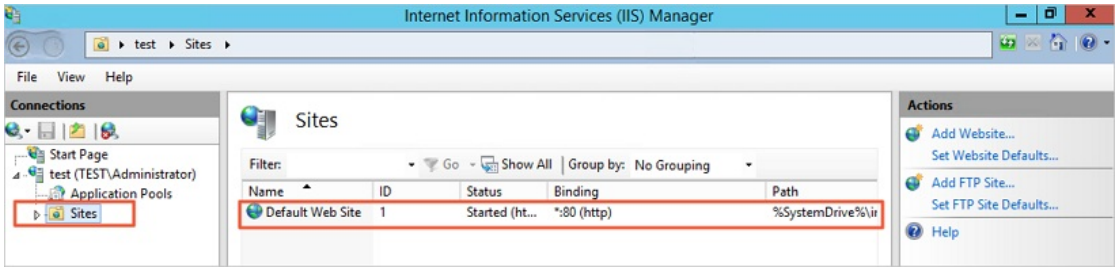
i. Find the Server Manager icon in the Windows taskbar and open Server Manager.



ii. In the top menu bar of the Server Manager window, choose **Tools > Internet Information Services (IIS) Manager**.

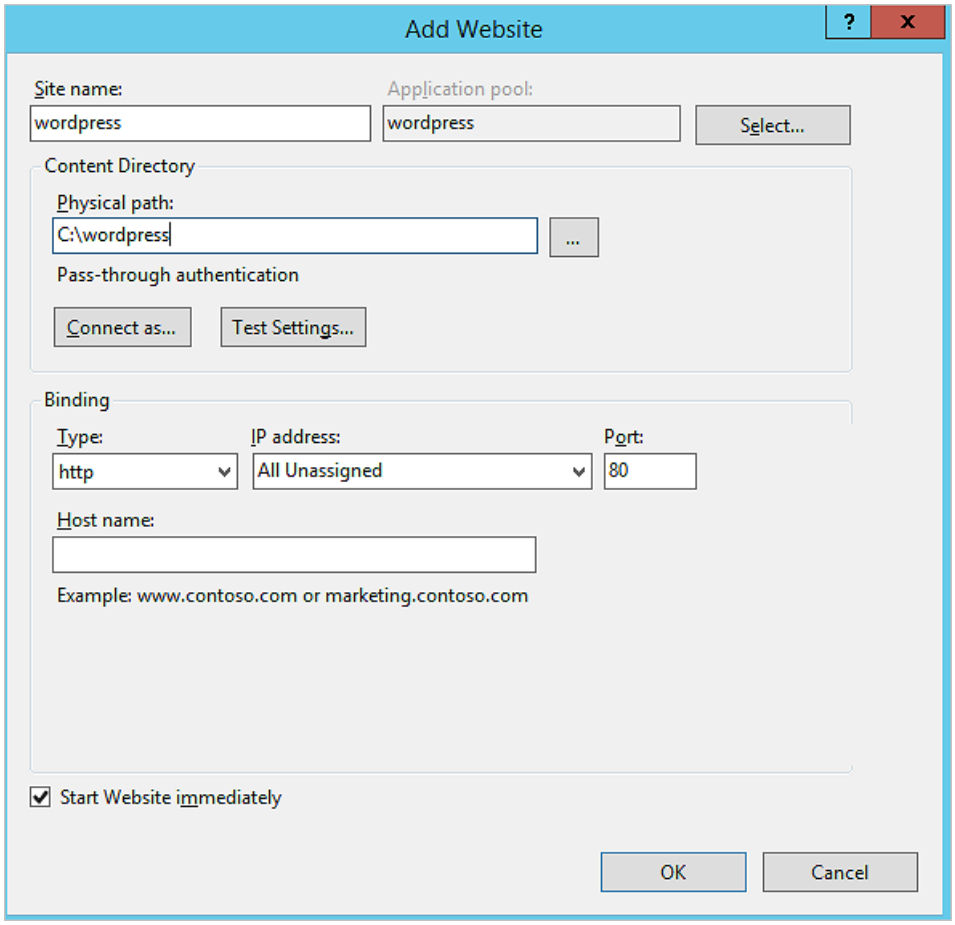
iii. In the **Connections** pane, choose **<ECS instance name> > Sites**.

iv. Delete the website that is bound to port 80, or change the port number from 80 to an unused port, such as port 8080.



v. In the right-side **Actions** pane, click **Add Website** to add the WordPress website.

The following figure shows an example.



The following content describes the parameters to be configured:

- Site name: the name of a custom website. In this tutorial, enter wordpress.
- Application pool: Select DefaultAppPool.
- Physical path: the directory where the WordPress installation package is decompressed. In this tutorial, select `C:\wordpress`.
- Port: Set it to 80.

5. Install WordPress and log on to the WordPress website.

i. Visit `http://localhost/` from the ECS instance. The WordPress installation page is displayed.

ii. Enter basic information of the website and click **Run the installation**.

The following content describes the parameters to be specified:

- Site Title: The name of the WordPress website. Example: demowp.
- Username: The username used to log on to WordPress. Keep your username secure. Example: testwp.
- Password: We recommend that you choose a secure password. Example: Wp.123456.
- Your Email: The email used to receive notifications. Example: 1234567890@aliyun.com.

iii. Click **Install WordPress**.

iv. Enter the username and password that are used to install WordPress and click **LOGIN**.

You are logged on to your WordPress website.

Resolve the domain name of the WordPress website

If you allow users to access your WordPress website by using the public IP address of the ECS instance, this compromises the security of the ECS instance. If you have a domain name or need to register a domain name for your WordPress website, perform the following steps. The domain name to register in this tutorial is `www.WordPress.EcsQuickStart.com`.

1. Register the domain name.

   For more information, see Register a generic domain nameHow to register an Alibaba Cloud domain name.

2. Apply for an ICP filing.

   If the website of your domain name is hosted on an ECS instance located in a mainland China region, you must apply for an ICP filing.

3. Resolve the domain name and bind it to the public IP address of the ECS instance.

   You must perform domain name resolution before you access your website by using a domain name. For more information, see Domain name resolution.

4. Return to the ECS instance on which the WordPress website is deployed. Go to the bin folder in the MySQL installation directory, right-click a blank area in this folder when you press and hold the `Shift` key, and then select **Open command window here**.

5. Log on to the MySQL database.

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

6. Use the wordpress database.

   ```
   use wordpress;
   ```

7. Replace `http://localhost/` with the new domain name.

   ```
   update wp_options set option_value = replace(option_value, 'http://localhost', 'http://www.WordPress.EcsQuickStart.com') where option_name = 'home' OR option_name = 'siteurl';
   ```

   The new domain name is configured for your WordPress website.

4.1.3. Manually build a WordPress website on an ECS instance that runs CentOS 8

WordPress is a blog-publishing system written in PHP. You can use WordPress as a content management system (CMS) or use WordPress to build your own websites on servers that support PHP and MySQL databases. This topic describes how to build a WordPress website on an Elastic Compute Service (ECS) instance that runs a Linux operating system.

Prerequisites

- A ECS Linux instance is created and an LNMP environment is manually built on the instance. For more information, see Manually build an LNMP environment on a CentOS 8 instance. In this topic, resources

of the following versions are used:

- Instance type: ecs.c6.large
- Operating system: CentOS 8.1 64-bit public image
- NGINX: 1.16.1
- MySQL: 8.0.17
- PHP:7.3.5
- WordPress: 5.4.2

> **Note** If you use software versions different from the preceding ones, you may need to adjust commands and parameter settings.

- An inbound rule is added to the security group of the instance to allow traffic on port 80. If you want to connect to the instance over SSH, another inbound rule is added to the security group to allows traffic on port 22. For more information, see Add a security group rule.

Context

The tutorial is intended for enterprises or individuals who are familiar with Linux, but new to building WordPress websites on Alibaba Cloud ECS instances. You can also use the WordPress image provided in Alibaba Cloud Marketplace to build a WordPress website.

Build a WordPress website

1. Use the ECS console to connect to the ECS Linux instance and configure a database for the WordPress website.
   i. Connect to the ECS instance.

      For more information, see Connect to a Linux instance by using a password.

   ii. (Optional)Change the CentOS 8 repository address.

      > **Note** If you have changed the CentOS 8 repository address when you bulid the LNMP environment, skip this step.

   iii. Log on to MySQL.

      Log on to MySQL by using the `root` username and entering the password that you set for the root user when you build the LNMP environment.

      ```
      mysql -uroot -p
      ```

   iv. Create a database for the WordPress website that you want to build.

      In this tutorial, the database created for the WordPress website is wordpress.

      ```
      create database wordpress;
      ```

v. Create a new user to manage the wordpress database to improve security.

In MySQL 5.7 and later, the password strength validation plug-in validate_password is installed by default. You can log on to MySQL to view the password strength rules.

```
show variables like "%password%";
```

In this tutorial, the created user is named `user` and its password is `PASSword123.`

```
create user 'user'@'localhost' identified by 'PASSword123.' ;
```

vi. Grant the user all permissions on the `wordpress` database.

```
grant all privileges on wordpress.* to 'user'@'localhost';
```

vii. Run the following command to validate the preceding configurations:

```
flush privileges;
```

viii. Run the following command to exit MySQL:

```
exit;
```

2. Download and depress WordPress and move it to the root directory of the WordPress website.

i. Go to the root directory of the NGINX website and download the WordPress package.

```
cd /usr/share/nginx/html
wget https://wordpress.org/wordpress-5.4.2.zip
```

ii. Decompress the WordPress package.

```
unzip wordpress-5.4.2.zip
```

iii. Copy the `wp-config-sample.php` file in the WordPress installation directory to `wp-config.php` and retain the `wp-config-sample.php` file.

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

iv. Edit the `wp-config.php` file.

```
vim wp-config.php
```

v. Press the /key to switch to the edit mode and modify MySQL-related configurations based on the wordpress database. The following content is an example of the modified code:

The data of the WordPress website will be saved in the wordpress database by the user named `user`.

```
// ** MySQL settings - The information is based on the host in use. ** //
/** The name of the WordPress database */
define('DB_NAME', 'wordpress');
/** The username of the MySQL database */
define('DB_USER', 'user');
/** The password of the MySQL database */
define('DB_PASSWORD', 'PASSword123.') ;
/** The host of the MySQL database */
define('DB_HOST', 'localhost');
```

vi. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

3. Modify the configuration file of NGINX.

i. Run the following command to open the NGINX configuration file:

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

ii. Press the /key to enter the edit mode.

Within the `location /` braces, replace the content that follows `root` with the root directory of the WordPress website. In this example, the root directory is */usr/share/nginx/html/wordpress*.

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

Within the `location ~ .php$` braces, replace the content that follows `root` with the root directory of the WordPress website.

```
location ~ \.php$ {
    root           /usr/share/nginx/html/wordpress;
    fastcgi_pass   unix:/run/php-fpm/www.sock;
    fastcgi_index  index.php;
    fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
    include        fastcgi_params;
}
```

Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and close the configuration file.

iii. Run the following command to restart the NGINX service:

```
systemctl restart nginx
```

4. Install WordPress and log on to the WordPress website.

i. On your local physical machine, use a browser to access `<The public IP address of the ECS instance>` to go to the WordPress installation page.

ii. Enter basic information of the website and click **Run the installation**.

The following content describes the parameters to the specified:

- Site Title: The name of the WordPress website. Example: demowp.
- Username: The username used to log on to WordPress. Keep your username secure. Example: testwp.
- Password: We recommend that you choose a secure password. Example: Wp.123456.
- Your Email: The email used to receive notifications. Example: 1234567890@aliyun.com.

iii. Click **Install Wordpress**.

iv. Enter the username `testwp` and password `Wp.123456` that are used to install WordPress, and then click **LOGIN**.

You are logged on to your WordPress website.

Resolve the domain name of the WordPress website

If you allow users to access your WordPress website by using the public IP address of the ECS instance, this compromises the security of the ECS instance. If you have a domain name or need to register a domain name for your WordPress website, perform the following steps. The domain name to register in this tutorial is `www.WordPress.EcsQuickStart.com`.

1. Register the domain name.

   For more information, see Register a generic domain nameHow to register an Alibaba Cloud domain name.

2. Apply for an ICP filing.

   If the website of your domain name is hosted on an ECS instance located in a mainland China region, you must apply for an ICP filing.

3. Resolve the domain name and bind it to the public IP address of the ECS instance.

   You must perform domain name resolution before you access your website by using a domain name. For more information, see Domain name resolution.

4. Return to the ECS console, connect to the ECS instance on which the WordPress website is deployed, and log on to the MySQL database.

   ```
   mysql -uroot -p
   ```

5. Use the wordpress database.

   ```
   use wordpress;
   ```

6. Replace the public IP address of the ECS instance with the new domain name.

   ```
   update wp_options set option_value = replace(option_value, 'http://<The public IP addre
   ss of the instance>', 'http://www.WordPress.EcsQuickStart.com') where option_name = 'ho
   me' OR option_name = 'siteurl';
   ```

7. Run the following command to exit MySQL:

   ```
   exit;
   ```

   The new domain name is configured for your WordPress website.

FAQ

- Problem description: After a static link is set for the WordPress website, the web page to which the static link points cannot be accessed.

  Solution: If you set a website to be pseudo-static, search engines can easily index the website. Before you set a static link for a WordPress website, you must specify pseudo-static rules in the NGINX server. Perform the following steps:

  i. Log on to the ECS instance on which the WordPress website is built.

  ii. Run the following command to open the NGINX configuration file:

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

  iii. Press the /key to enter the edit mode. Within the `location /` braces, add the following code:

  ```
  if (-f $request_filename/index.html){
  rewrite (.*) $1/index.html break;
  }
  if (-f $request_filename/index.php){
  rewrite (.*) $1/index.php;
  }
  if (!-f $request_filename){
  rewrite (.*) /index.php;
  }
  ```

  Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

  iv. Run the following command to restart NGINX:

  ```
  systemctl restart nginx
  ```

- Problem description: When I update the version of WordPress or upload a topic or plug-in in WordPress, a message is displayed and indicates that an FTP logon credential is required or that the directory cannot be created.

  Solution:

  i. Log on to the ECS instance on which the WordPress website is built.

  ii. Run the following command to open the configuration file of WordPress:

  ```
  vim /usr/share/nginx/html/wordpress/wp-config.php
  ```

  iii. Press the /key to enter the edit mode. Add the following code to the end of the file:

  ```
  define("FS_METHOD","direct");
  define("FS_CHMOD_DIR", 0777);
  define("FS_CHMOD_FILE", 0777);
  ```

  Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

  iv. Return to the WordPress dashboard and refresh the page. Check whether the FTP logon credential issue is solved.

If the issue that the directory cannot be created persists, return to the ECS instance and run the following command to change the owner of the root directory of the WordPress website to a NGINX user. In this example, the NGINX user is the `nginx` user.

```
chown -R nginx /usr/share/nginx/html/wordpress
```

4.1.4. Manually build a WordPress website on a CentOS 7 ECS instance

WordPress is a blog-publishing system written in PHP. You can use WordPress as a content management system (CMS) or use WordPress to build your own websites on servers that support PHP and MySQL databases. This topic describes how to build a WordPress website on an ECS instance that runs a Linux distribution.

Prerequisites

-
- A VPC-type security group is created. Inbound rules are added to the security group to allow traffic on port 80. If you want to connect to the Linux instance by using SSH, you must also allow traffic on port 22 in the inbound rules. For information about how to add security group rules, see Add a security group rule.
- A Linux ECS instance is created and deployed with the LNMP environment. For more information, see Manually build an LNMP environment on a CentOS 7 instance. Resources of the following versions are used in this topic:
  - Instance type: ecs.c6.large
  - Operating system: a CentOS 7.2 64-bit public image
  - NGINX: 1.16.1
  - MySQL: 5.7.29
  - PHP: 7.0.33
  - WordPress: 5.1.6

> **Note** If you use resources that are of different versions from the preceding ones, you may need to adjust commands and parameter settings.

Context

The tutorial is intended for enterprises or individuals who are familiar with Linux, but new to building WordPress websites on Alibaba Cloud ECS instances. You can also use the WordPress image provided in Alibaba Cloud Marketplace to build a WordPress website.

Build a WordPress website

1. Use the ECS console to connect to the Linux instance that is deployed with an LNMP environment and configure a database for the WordPress website.
   i. Connect to the ECS instance.

      For more information, see Connect to a Linux instance by using a password.

ii. Log on to MySQL.

Log on to MySQL by using the `root` username and entering the password that you set for the root user when you build the LNMP environment.

```
mysql -uroot -p
```

iii. Create a database for the WordPress website that you want to build.

In this tutorial, the database created for the WordPress website is wordpress.

```
create database wordpress;
```

iv. Create a new user to manage the wordpress database to improve security.

In MySQL 5.7 and later, the password strength validation plug-in validate_password is installed by default. You can log on to MySQL to view the password strength rules.

```
show variables like "%password%";
```

In this tutorial, the created user is named `user` and its password is `PASSword123.`

```
create user 'user'@'localhost' identified by 'PASSword123.' ;
```

v. Grant the user all permissions on the wordpress database.

```
grant all privileges on wordpress.* to 'user'@'localhost' identified by 'PASSword12
3.' ;
```

vi. Run the following command to validate the preceding configurations:

```
flush privileges;
```

vii. Run the following command to exit MySQL:

```
exit;
```

2. Download WordPress and move it to the root directory of the website to be built.

i. Download WordPress.

Run the following **yum** command to download WordPress. The downloaded WordPress package is automatically saved in the */usr/share/wordpress* directory.

```
yum -y install wordpress
```

ii. Move the WordPress package to the root directory of the website to be built.

```
mv /usr/share/wordpress /usr/share/nginx/html/wordpress
```

3. Modify the configuration file of WordPress.

i. Go to the new WordPress path and connect to the *wp-config.php* configuration file by using the soft link in the path.

```
cd /usr/share/nginx/html/wordpress
ln -snf /etc/wordpress/wp-config.php wp-config.php
```

ii. Edit the `wp-config.php` file.

```
vim wp-config.php
```

iii. Press the /key to switch to the edit mode and modify MySQL-related configurations based on the wordpress database. The following content is an example of the modified code:

The data of the WordPress website will be saved in the wordpress database by the user named `user`.

```
// ** MySQL settings - The information is based on the host in use. ** //
/** The name of the WordPress database */
define('DB_NAME', 'wordpress');
/** The username of the MySQL database */
define('DB_USER', 'user');
/** The password of the MySQL database */
define('DB_PASSWORD', 'PASSword123.') ;
/** The host of the MySQL database */
define('DB_HOST', 'localhost');
```

iv. Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

4. Modify the configuration file of NGINX.

i. Run the following command to open the configuration file of NGINX:

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

ii. Press the /key to enter the edit mode.

Within the `server` braces, replace the content after `root` with the root directory of the WordPress website. The root directory is */usr/share/nginx/html/wordpress* in this example.

```
server {
    listen       80 default_server;
    listen       [::]:80 default_server;
    server_name  _;
    root         /usr/share/nginx/html/wordpress;
```

Within the `location ~ .php$` braces, replace the content after `root` with the root directory of the WordPress website.

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

Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

iii. Run the following command to restart the NGINX service:

```
systemctl restart nginx
```

5. Install WordPress and log on to the WordPress website.
   i. On your local physical machine, use a browser to access `<The public IP address of the ECS instance>` to go to the WordPress installation page.
   ii. Enter basic information of the website and click **Run the installation**.

   The following content describes the parameters to the specified:

   - Site Title: The name of the WordPress website. Example: demowp.
   - Username: The username used to log on to WordPress. Keep your username secure. Example: testwp.
   - Password: We recommend that you choose a secure password. Example: Wp.123456.
   - Your Email: The email used to receive notifications. Example: 1234567890@aliyun.com.

   iii. Click **Install Wordpress**.
   iv. Enter the username `testwp` and password `Wp.123456` that are used to install WordPress, and then click **LOGIN**.

   You are logged on to your WordPress website.

Resolve the domain name of the WordPress website

If you allow users to access your WordPress website by using the public IP address of the ECS instance, this compromises the security of the ECS instance. If you have a domain name or need to register a domain name for your WordPress website, perform the following steps. The domain name to register in this tutorial is `www.WordPress.EcsQuickStart.com`.

1. Register the domain name.

   For more information, see Register a generic domain nameHow to register an Alibaba Cloud domain name.

2. Apply for an ICP filing.

   If the website of your domain name is hosted on an ECS instance located in a mainland China region, you must apply for an ICP filing.

3. Resolve the domain name and bind it to the public IP address of the ECS instance.

   You must perform domain name resolution before you access your website by using a domain name. For more information, see Domain name resolution.

4. Return to the ECS console, connect to the ECS instance on which the WordPress website is deployed, and log on to the MySQL database.

   ```
   mysql -uroot -p
   ```

5. Use the wordpress database.

   ```
   use wordpress;
   ```

6. Replace the public IP address of the ECS instance with the new domain name.

   ```
   update wp_options set option_value = replace(option_value, 'http://<The public IP addre
   ss of the instance>', 'http://www.WordPress.EcsQuickStart.com') where option_name = 'ho
   me' OR option_name = 'siteurl';
   ```

7. Run the following command to exit MySQL:

```
exit;
```

The new domain name is configured for your WordPress website.

FAQ

- Problem description: After a static link is set for the WordPress website, the web page to which the static link points cannot be accessed.

  Solution: If you set a website to be pseudo-static, search engines can include the website more easily. Before you set a static link for a WordPress website, you must specify pseudo-static rules in the NGINX server. Perform the following steps:

  i. Log on to the ECS instance on which the WordPress website is built.

  ii. Run the following command to open the configuration file of NGINX:

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

  iii. Press the /key to enter the edit mode. Within the `location /` braces, add the following code:

  ```
  if (-f $request_filename/index.html){
  rewrite (.*) $1/index.html break;
  }
  if (-f $request_filename/index.php){
  rewrite (.*) $1/index.php;
  }
  if (! -f $request_filename){
  rewrite (.*) /index.php;
  }
  ```

  Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

  iv. Run the following command to restart the NGINX service:

  ```
  systemctl restart nginx
  ```

- Problem description: When I update the version of WordPress or upload a topic or plug-in in WordPress, a message is displayed. The message indicates that an FTP logon credential is required or that the directory cannot be created.

  Solution:

  i. Log on to the ECS instance on which the WordPress website is built.

  ii. Run the following command to open the configuration file of WordPress:

  ```
  vim /usr/share/nginx/html/wordpress/wp-config.php
  ```

  iii. Press the /key to enter the edit mode. At the bottom, add the following code:

  ```
  define("FS_METHOD","direct");
  define("FS_CHMOD_DIR", 0777);
  define("FS_CHMOD_FILE", 0777);
  ```

  Press the *Esc* key to exit the edit mode. Enter `:wq` and press the Enter key to save and exit the configuration file.

iv. Return to the WordPress dashboard and refresh the page. Check whether the problem that an FTP logon credential is required is solved.

If the problem that the directory cannot be created persists, return to the ECS instance and run the following command to change the owner of the root directory of the WordPress website to a NGINX user. In this example, the NGINX user is the `nginx` user.

```
chown -R nginx /usr/share/nginx/html/wordpress
```

4.2. Build a Drupal website

4.2.1. Build a Drupal website based on an Alibaba Cloud Marketplace image

This topic describes how to build a Drupal e-commerce website on an Elastic Compute Service (ECS) instance that uses an Alibaba Cloud Marketplace image.

Context

Drupal is an open source content management framework (CMF) written in PHP. Drupal consists of a content management system (CMS) and a PHP development framework. You can use Drupal to build dynamic websites that provide various features and services. Drupal is commonly used in a variety of applications from personal blogs to large communities.

The procedure described in this topic is applicable to users who are familiar with Linux operating systems, but are new to web development on Alibaba Cloud ECS instances and want to build a website in a quick manner.

In this example, the following operating system and software versions are used:

- Operating system: CentOS 7.3 64-bit
- Apache 2.4.45
- MySQL 5.6.36
- PHP 5.6.30
- Drupal 8.3.4

Procedure

To build a Drupal website based on an Alibaba Cloud Marketplace image, perform the following steps:

1. Step 1: Create an ECS instance
2. Step 2: Select a Drupal website image
3. Step 3: Install Drupal

Step 1: Create an ECS instance

Create an ECS instance to build a small-sized website for personal use. Then, you can upgrade the configurations of the instance or optimize the architecture as your requirements increase. In this example, an ECS instance of the ecs.c6.large instance type is used.

Step 2: Select a Drupal website image

1.
2.
3.
4. On the **Instances** page, click **Create Instance**.
5. In the **Image** section of the **Custom Launch** tab, choose **Marketplace Image > Select from Alibaba Cloud Marketplace (including operating system)**.
6. Enter `Drupal` in the search box and click **Search** to select a Drupal website image.
7. Click **Use**.
8. On the **Custom Launch** tab, you can see that the Alibaba Cloud Marketplace image that you selected is displayed in the **Image** section. Assign a public IP address to the instance and complete other settings to create the instance. For more information, see Create an instance by using the wizard.

Step 3: Install Drupal

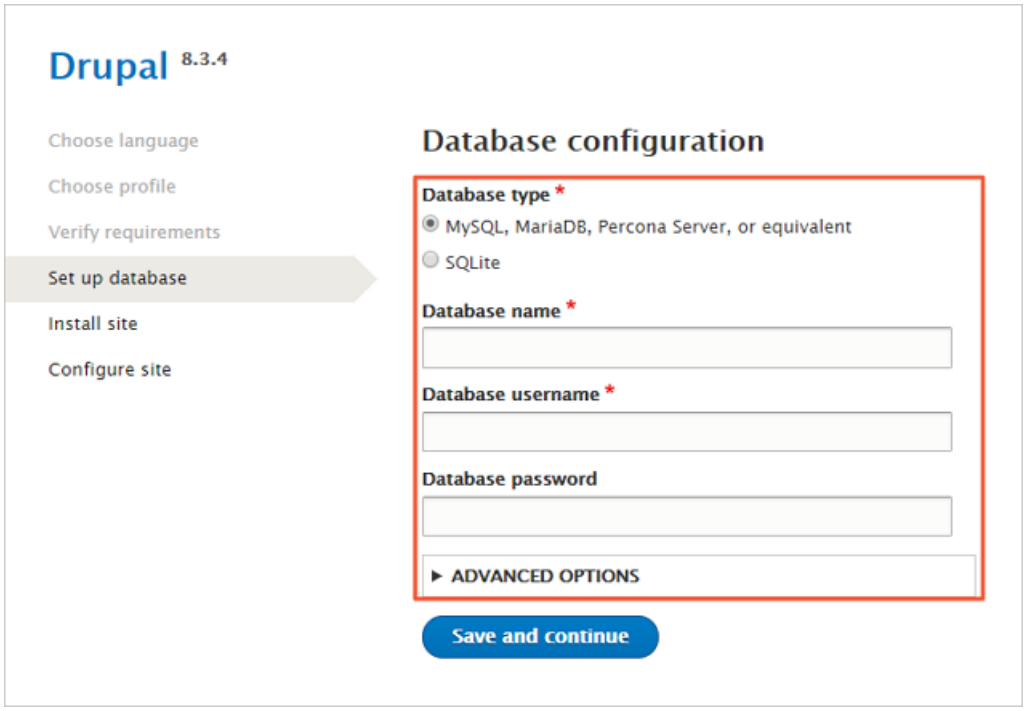
1. Access `http://<Public IP address of the instance>/phpMyAdmin` by using your browser.

   You can obtain the public IP address of the instance from the IP Address column corresponding to the instance on the Instances page in the ECS console.
2. Use the username and password of MySQL provided by the Alibaba Cloud Marketplace image to log on to phpMyAdmin.
3. In the left-side navigation pane, click **NEW**. In the top navigation bar, click **SQL**.
4. Create a database and user for Drupal.

   Enter the following SQL statements in the field and configure the following parameters:
   - drupalDBName: Specify a name for the database.
   - UserName: Specify a user for the database.
   - IP: Enter localhost or 127.0.0.1 if you perform the operations on your computer.
   - UserPassWord: Specify a password for the database.

   ```
   CREATE DATABASE drupalDBName;
   CREATE USER UserName;
   GRANT ALL PRIVILEGES ON *.* TO 'UserName'@'IP' IDENTIFIED BY 'UserPassWord' WITH GRANT
   OPTION;
   FLUSH PRIVILEGES;
   ```
5. Click **Go**.
6. Access `http://<Public IP address of the instance>/drupal` by using your browser to go to the Drupal installation page. Select an installation language from the Choose language drop-down list and click **Save and continue**.

7. Select the standard installation method and click **Save and continue**.
8. Enter the information of the created database and click **Save and continue**.



9. After Drupal is installed, enter the site information on the website setting page and click **Save and continue**.
   Then, you can log on to the Drupal website to customize the settings.

4.2.2. Manually build a Drupal website

This topic describes how to use Drupal to deploy an e-commerce website on a CentOS 7 ECS instance.

Prerequisites

-
- An ECS instance that has a public IP address is created and deployed with a LAMP environment. For more information, see Build a LAMP environment on a CentOS 7 instance.

Context

Drupal is an open source content management framework (CMF) written in PHP. Drupal consists of a content management system (CMS) and a PHP development framework. You can use Drupal to build dynamic websites that provide various features and services. Drupal is commonly used in a variety of applications, from personal blogs to large communities.

This topic is intended for users who are familiar with Linux, but new to web development on Alibaba Cloud ECS instances. You can also build a Drupal website based on an Alibaba Cloud Marketplace image. For more information, see Build a Drupal website based on an Alibaba Cloud Marketplace image.

Configuration

The following instance configurations and software versions are used in the example. The operations may vary depending on your instance configurations and software versions.

- Instance type: ecs.c6.large
- Operating system: CentOS 7.8 64-bit
- Apache HTTP Server: 2.4.6
- MySQL: 5.7.31
- PHP: 7.0.33
- Drupal: 8.1.1

Configure the database information

1. Use a local browser to access *http://<Public IP address of the instance>/phpMyAdmin*.
2. Use the username and password of a MySQL database to log on to phpMyAdmin.
3. At the top of the page, click **SQL**.
4. Create a database and user for Drupal.

   Enter the following SQL statement in the editor:

   ```
   CREATE DATABASE <DrupalDBName>;
   CREATE user '<UserName>'@'<IP>' IDENTIFIED BY '<UserPassWord>';
   FLUSH PRIVILEGES;
   ```

   Specify the parameters in the SQL statement:

   - `<DrupalDBName>`: Specify a name for the database.
   - `<UserName>`: Specify a user for the database.
   - `<IP>`: Specify the IP address of the local host or 127.0.0.1.
   - `<UserPassWord>`: Specify a password for the database.

     > **Note** You can execute the `show variables like 'validate_password%';` SQL statement to query the password strength rules for the database.

5. Click **Go**.

Install Drupal

1. Connect to an ECS instance in which an LAMP environment is deployed.

For more information about the remote connection methods, see Connect to a Linux instance by using a password.

2. Download and configure Drupal.
   i. Download the Drupal installation package.

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

   ii. Decompress the Drupal installation package and move the installation files to the root directory of your Apache website.

```
yum install unzip -y
unzip drupal-8.1.1.zip
```

```
mv drupal-8.1.1/* /var/www/html
```

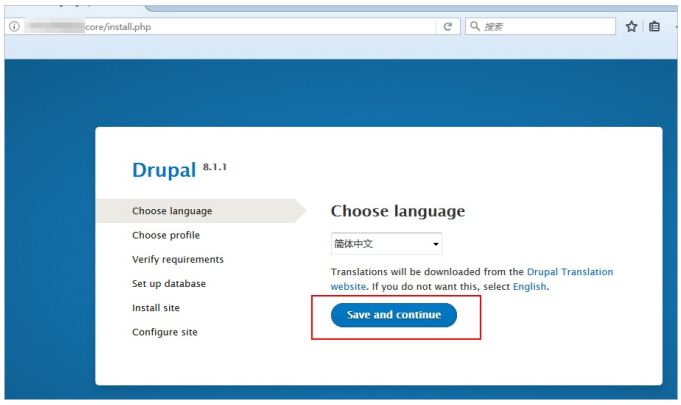
   iii. Modify the owner and group of the *sites* directory.

```
chown -R daemon:daemon /var/www/html/sites
```

   iv. Restart the Apache service.

```
systemctl restart httpd
```

3. Use a browser to access the website and install Drupal.
   i. User a local browser to access *<Public IP address of the ECS instance>* and go to the Drupal installation page. Select the installation language from the Choose language drop-down list, and then click **Save and continue**.



   ii. Select Standard, and click **Save and continue**.
   iii. Enter the information of the configured database, click **Save and continue**.
   iv. After the installation is complete, go to the website settings page, enter website information, and then click **Save and continue**.

What's next

After the installation is complete, you can customize your website pages.

4.3. Build multiple Web sites

4.3.1. Build multiple websites on a Windows instance

This topic describes how to use Internet Information Services (IIS) to build multiple websites on a Windows Elastic Compute Service (ECS) instance. In this topic, the instance runs Windows Server 2012 R2 64-bit.

Prerequisites

-
- An instance is created, and the web environment that consists of IIS, PHP, and MySQL is deployed on the instance. You can use a Windows image from Alibaba Cloud Marketplace that is installed with the environment to deploy the environment.

Context

This tutorial is intended for users who are familiar with Windows and want to optimize the O&M process by making efficient use of resources and managing sites in a centralized manner. For example, you can configure multiple blogging platforms of different categories or build multiple websites to handle complex business tasks on an instance.

In this tutorial, IIS is used to simultaneously build the `windows-testpage-1` and `windows-testpage-2` websites and configure different domain names on the same port to access the websites.

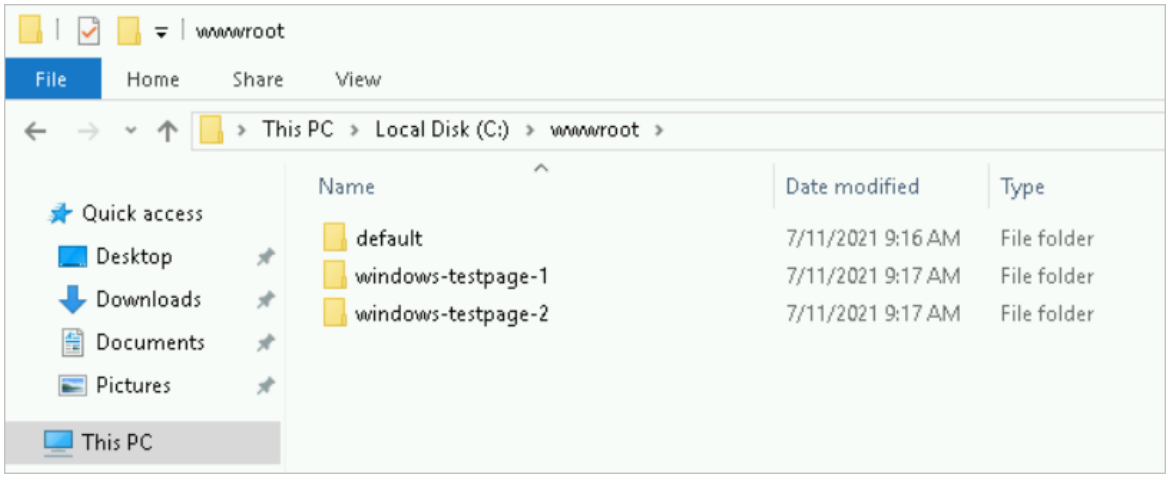
In this topic, the following instance configurations are used:

- Instance type: ecs.c6.large
- Operating system: Windows Server 2012 R2 64-bit

Create test websites

1. Connect to the instance on which the web environment is deployed.

   For more information, see Connect to a Windows instance by using a password.
2. On the desktop, click **This PC** and go to the *C:\wwwroot* path in the default root directory.
3. Create the `windows-testpage-1` and `windows-testpage-2` folders.



4. Open the `windows-testpage-1` folder, create the *test1.php* file in the folder, and then enter the

following content in the file:

```
<?php
echo "<title>Test-1</title>";
echo "windows-test-1";
?>
```

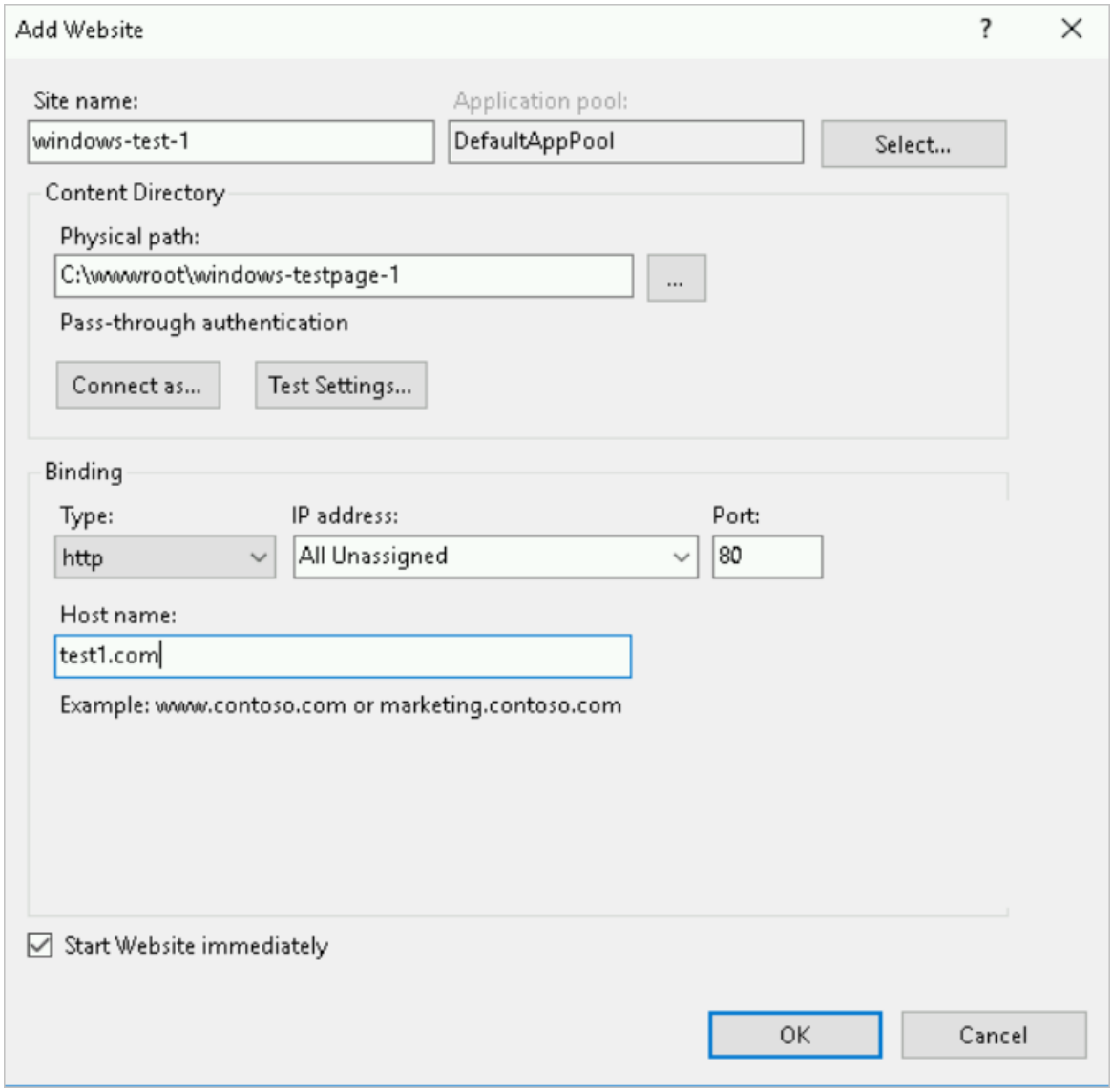
5. Open the `windows-testpage-2` folder, create the *test2.php* file in the folder, and then enter the following content in the file:

```
<?php
echo "<title>Test-2</title>";
echo "windows-test-2";
?>
```

Configure IIS

1. In the taskbar, click the **Server Manager** icon  .
2. In the top navigation bar, choose **Tools > Internet Information Services (IIS) Manager**.
3. In the left-side navigation pane of IIS Manager, click the name of the server and click **Sites**.
4. In the Actions section on the right, click **Add Website**. Add the `windows-testpage-1` website and click **OK**.

   The following figure shows how to configure the website.

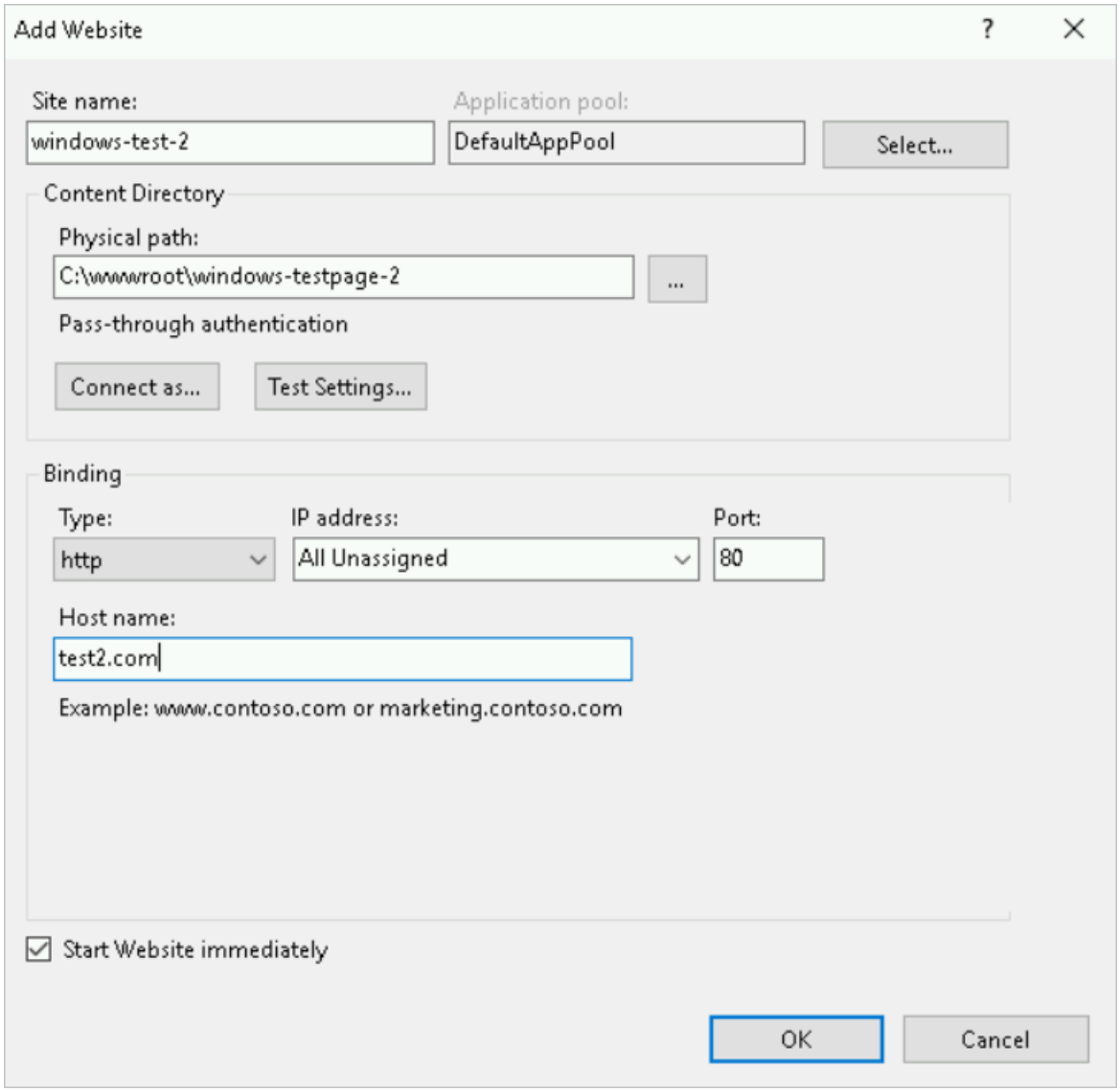


The following list describes how to configure the parameters:

- Site name: Enter `windows-testpage-1`.
- Application pool: Select DefaultAppPool.
- Physical path: Select a physical path for the `windows-testpage-1` website.
- Host name: Specify the `test1.com` domain name as the hostname.

5. In the Actions section on the right, click **Add Website**. Add the `windows-testpage-2` website and click **OK**.

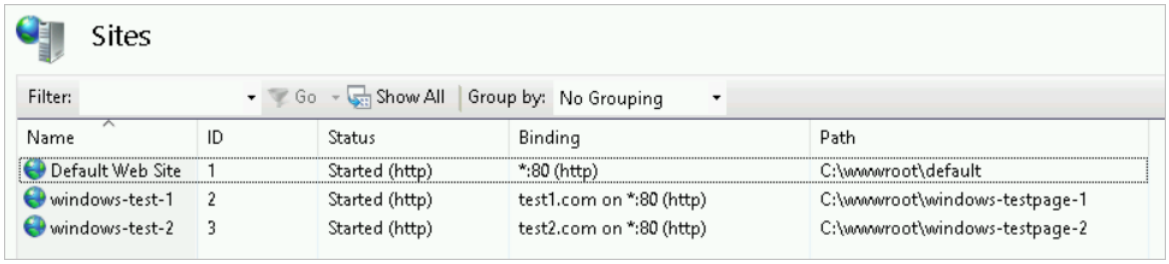
   The following figure shows how to configure the website.



The following list describes how to configure the parameters:

- Site name: Enter `windows-testpage-2`.
- Application pool: Select DefaultAppPool.
- Physical path: Select a physical path for the `windows-testpage-2` website.
- Host name: Specify the `test2.com` domain name as the hostname.

The following figure indicates that the websites are added.



(Optional) Configure the hosts file on the local host

In this tutorial, the domain names are used only for test purposes. You must configure IP mapping in the local hosts file. If you use the actual server domain names when you configure the websites, skip this step. In this tutorial, the local physical server uses the Windows operating system.

1. Go to the *C:\Windows\System32\drivers\etc* directory.
2. Copy the *hosts* file for backup.

   Retain the *hosts - copy* file, which can be used to restore the *hosts* file to the initial state after the test is complete.
3. Modify the *hosts* file.

   Append the following content to the end of the file, save the file, and then exit the file:

   ```
   <Public IP address of the instance> test1.com
   <Public IP address of the instance> test2.com
   ```
4. Go back to the Windows desktop and press *Win+R*.
5. In the **Run** dialog box, enter *cmd* and click **OK**.
6. Run the following command in the command line to make the configurations of the *hosts* file immediately take effect:

   ```
   ipconfig /flushdns
   ```

Result

You can access the two test websites from a browser on the local host.

- Enter `test1.com/test1.php` in the address bar of the browser and press the Enter key. The `windows-testpage-1` website is displayed, as shown in the following figure.

- Enter `test2.com/test2.php` in the address bar of the browser and press the Enter key. The `windows-testpage-2` website is displayed, as shown in the following figure.

Multiple websites are built. In your actual operations, you need only to make sure that the hostnames and project paths are correctly configured to access your websites. You can install SSL certificates in IIS. For more information, see Install SSL certificates on IIS servers.

4.3.2. Build multiple websites in CentOS 7

This topic describes how to use NGINX to build multiple websites on an ECS instance that runs CentOS 7.

Prerequisites

-
- An ECS instance that has a public IP address is created and deployed with an LNMP (Linux, NGINX, MySQL, and PHP) environment. For more information, see Manually build an LNMP environment on a CentOS 7 instance.

Context

This tutorial is intended for users who are familiar with Linux and want to improve O&M efficiency by making efficient use of resources and managing sites in a centralized manner. For example, you can configure multiple blogging platforms of different categories or build multiple websites for sophisticated businesses on an instance.

In this tutorial, the *Testpage-1* and *Testpage-2* sites are simultaneously built on an instance deployed with an LNMP environment and then accessed.

The following instance configurations are used in the example:

- Instance type: ecs.c6.large
- Operating system: CentOS 7.8 64-bit

Create test sites

1. Connect to the instance that is deployed with an LNMP environment.

   For more information, see Connect to a Linux instance by using a password.

2. Run the following command to go to the configured website root directory:

   ```
   cd /usr/share/nginx/html
   ```

3. Run the following commands to create two test folders.

   The folders are used to store information of the test websites, which is the project code.

   ```
   mkdir Testpage-1
   mkdir Testpage-2
   ```

4. Configure information of *Testpage-1*.

   i. Run the following command to go to *Testpage-1*:

   ```
   cd /usr/share/nginx/html/Testpage-1/
   ```

   ii. Run the following command to create and edit the `index.html` file:

   ```
   vim index.html
   ```

   iii. Press the /key to switch to the edit mode and enter the following test content:

   ```
   Test page 1
   ```

   Press the *Esc* key, enter *:wq*, and then press the Enter key to save the file and exit the edit mode.

5. Configure information of *Testpage-2*.

i. Run the following command to go to *Testpage-2*:

```
cd /usr/share/nginx/html/Testpage-2/
```

ii. Run the following command to create and edit the `index.html` file:

```
vim index.html
```

iii. Press the /key to switch to the edit mode and enter the following test content:

```
Test page 2
```

Press the *Esc* key, enter *:wq*, and then press the Enter key to save the file and exit the edit mode.

Configure NGINX

1. Run the following command to check the *nginx.conf* configuration file:

```
cat /etc/nginx/nginx.conf
```

View the `include` configuration information in the `http{}` module.

`include /etc/nginx/conf.d/*.conf;` indicates that NGINX will obtain site information from all files in the *.conf* format in this path, as shown in the following figure.

```
http {
    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;
    tcp_nodelay         on;
    keepalive_timeout   65;
    types_hash_max_size 2048;

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

    # Load modular configuration files from the /etc/nginx/conf.d directory.
    # See http://nginx.org/en/docs/ngx_core_module.html#include
    # for more information.
    include /etc/nginx/conf.d/*.conf;

    server {
        listen       80 default_server;
        listen       [::]:80 default_server;
        server_name  _;
        root         /usr/share/nginx/html;
```

2. Run the following command to go inside the */etc/nginx/conf.d* path:

```
cd /etc/nginx/conf.d
```

3. Create and configure the NGINX configuration file for *Testpage-1*.
   i. Run the following command to create and edit the configuration file:

   ```
   vim Testpage1.conf
   ```

   ii. Press the /key to switch to the edit mode and enter the following content.

   For the commented content, replace the server domain name and the project path with your actual parameter values.

   ```
   server {
       listen       80;
       server_name  testpage1.com;    #The test domain name is used here. Use the doma
   in name of your server in the actual configuration.
       #charset koi8-r;
       access_log  /var/log/nginx/b.access.log  main;
       location / {
           root   /usr/share/nginx/html/Testpage-1;    #The test site path is used her
   e. Use the path of your project code in the actual configuration.
           index  index.html index.htm;
       }
       #error_page  404              /404.html;
       error_page   500 502 503 504  /50x.html;
       location = /50x.html {
           root   /usr/share/nginx/html;
       }
   }
   ```

   Press the *Esc* key, enter *:wq*, and then press the Enter key to save the file and exit the edit mode.

4. Create and configure the NGINX configuration file for *Testpage-2*.
   i. Run the following command to create and edit the configuration file:

   ```
   vim Testpage2.conf
   ```

ii. Press the /key to switch to the edit mode and enter the following content.

For the commented content, replace the server domain name and the project path with your actual parameter values.

```
server {
    listen       80;
    server_name  testpage2.com;    #The test domain name is used here. Use the doma
in name of your server in the actual configuration.
    #charset koi8-r;
    access_log  /var/log/nginx/b.access.log  main;
    location / {
        root   /usr/share/nginx/html/Testpage-2;    #The test site path is used her
e. Use the path of your project code in the actual configuration.
        index  index.html index.htm;
    }
    #error_page  404              /404.html;
    error_page   500 502 503 504  /50x.html;
    location = /50x.html {
        root   /usr/share/nginx/html;
    }
}
```

Press the *Esc* key, enter *:wq*, and then press the Enter key to save the file and exit the edit mode.

5. Run the following command to restart the NGINX server:

```
systemctl restart nginx
```

(Optional) Configure the hosts file on the local host

You must configure IP mapping in the local hosts file because all of the information used in this tutorial is test values. If you use the actual server domain name when you configure the sites, skip this step. In this tutorial, the local physical machine uses the Windows operating system.

1. Go to the *C:\Windows\System32\drivers\etc* directory.
2. Copy the *hosts* file for backup.

   Keep the *hosts - copy* file to restore the *hosts* file to its initial status after the configuration is completed.
3. Modify the *hosts* file.

   Append the following content to the end of the file:

   ```
   <The public IP address of the instance> testpage1.com
   <The public IP address of the instance> testpage2.com
   ```

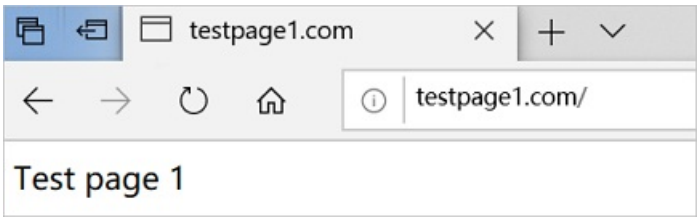
   Save the file and exit.
4. Go back to the Windows desktop and press *Win+R*.
5. In the **Run** dialog box that appears, enter *cmd* and click **OK**.
6. Run the following command in the command line to immediately apply the configurations of the *hosts* file:

   ```
   ipconfig /flushdns
   ```

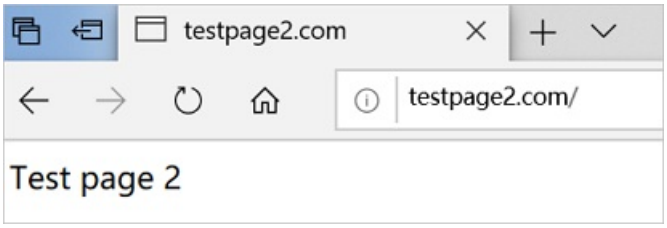
Result

You can access the two test sites from a browser on the local host.

- If you go to `testpage1.com/`, you can view the content of the *Testpage-1* site, as shown in the following figure.



- If you go to `testpage2.com/`, you can view the content of the *Testpage-2* site, as shown in the following figure.



Multiple websites have been built. In your actual operation, you only need to make sure that the domain names and project paths are correctly configured, and then you can access these websites.

4.4. Build a Magento e-commerce website on ECS

4.4.1. Build a Magento e-commerce website on an ECS instance

Magento is an open source e-commerce solution that has a modular architecture and varied expansion features. Magento supports PHP versions from 5.6 to 7.1. It uses MySQL databases to store data. This topic describes how to build a Magento e-commerce website on an Elastic Compute Service (ECS) instance that runs a CentOS 7 operating system

Prerequisites

-
- Inbound rules are added to the security group of the ECS instance to allow traffic on ports 80 and 3306. For more information, see Add a security group rule.

> **Note** Most clients are located in LANs and can map their private IP addresses to public IP addresses to communicate with external resources. Therefore, the IP addresses returned by the **ipconfig** or **ifconfig** command may not be the actual public IP addresses of the clients. If clients cannot access the Magento website after it is built, verify the public IP addresses of the clients.

Rule direction	Action	Protocol type	Port range	Authorization type	Authorization object
Inbound	Allow	HTTP (80)	80/80	IPv4 CIDR block	The CIDR blocks containing the public IP addresses of all clients that need to access the Magento website. Separate the CIDR blocks with commas (,). To allow all clients to access the Magento website, specify 0.0.0.0/0 as an authorization object.
Inbound	Allow	MySQL (3306)	3306/3306	IPv4 CIDR block	The CIDR blocks containing the public IP addresses of all clients that need to access the MySQL database services. Separate the CIDR blocks with commas (,). To allow all clients to access the services, specify 0.0.0.0/0 as an authorization object.

Context

In this topic, an ECS instance that has the following configurations is used:

- Instance type: ecs.c6.large
- Operating system: CentOS 7.2 64-bit public image
- CPU: 2 vCPUs
- Memory: 4 GiB

> **Note** If you want to build a Magento server, the memory of the selected instance type must be at least 2 GiB.

- Network type: Virtual Private Cloud (VPC)
- IP address: public IP address

In the sample procedure, the following software versions are used:

- Apache HTTP Server: 2.4.6
- MySQL: 5.7
- PHP: 7.0
- Composer: 1.8.5
- Magento: 2.1

If you use software versions different from the preceding ones, you may need to adjust commands and parameter settings.

Procedure

Perform the following steps to build a Magento e-commerce website on an Alibaba Cloud ECS instance:

Step 1. Install and configure Apache HTTP Server

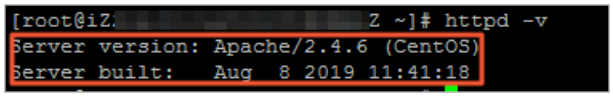
1. Install Apache.
   i. Run the following command to install Apache:

   ```
   yum install httpd -y
   ```

   ii. Run the following command to check whether Apache is installed:

   ```
   httpd -v
   ```

   The following command output indicates that Apache is installed.

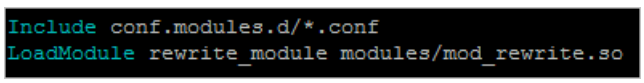
   

2. Configure Apache.
   i. Run the following command to open the configuration file of Apache:

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

   ii. Add `LoadModule rewrite_module modules/mod_rewrite.so` below `Include conf.modules.d/*.conf`. Perform the following operations:
      a. Move the pointer to the beginning of the line below the `Include conf.modules.d/*.conf` line.
      b. Press the /key to enter the edit mode.
      c. Enter `LoadModule rewrite_module modules/mod_rewrite.so`.

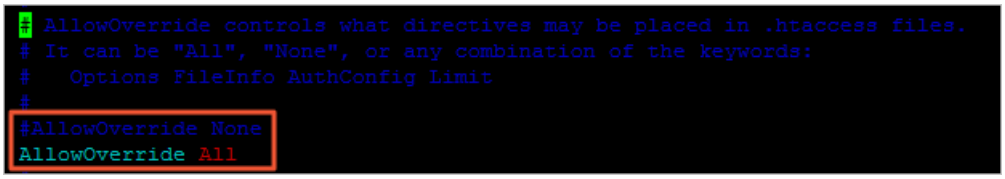
   The following figure shows the added content.

iii. Replace `AllowOverride None` in the following content with `AllowOverride All`.

```
# 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
#
# Add a number sign (#) at the beginning of the line to comment out this line.
#AllowOverride None
# Add the following content:
AllowOverride All
```

The following figure shows the replacement result.



iv. Press the *Esc* key, enter *:wq*, and then press the Enter key to save and close the configuration file.

3. Run the following command to start Apache:

```
systemctl start httpd
```

4. Run the following command to configure Apache to run upon system startup:

```
systemctl enable httpd
```

Step 2. Install and configure MySQL

1. Install MySQL on the ECS instance.

i. Run the following command to add a MySQL YUM repository:

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

ii. Run the following command to install MySQL:

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

2. Run the following command to start MySQL:

```
systemctl start mysqld
```

3. Run the following command to enable MySQL to run on system startup:

```
systemctl enable mysqld
```

4. Configure MySQL.

i. Run the following command to check the */var/log/mysqld.log* file and obtain and record the initial password of the root user:

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

The following command output is returned:

```
2016-12-13T14:57:47.535748Z 1 [Note] A temporary password is generated for root@loc
alhost: p0/G28g>lsHD
```

> **Note** This initial password is used when you reset the password of the root user.

ii. Run the following command to configure the security settings of MySQL:

```
mysql_secure_installation
```

Perform the following operations:

a. Set the password of the root user.

```
Enter password for user root: # Enter the initial password that you obtained in
the previous step.
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 #
Enter Y to change the password of the root user.
New password: # Enter a new password that is 8 to 30 characters in length. The
password must contain uppercase letters, lowercase letters, digits, and special
characters. Supported special characters include ( ) `  ~ ! @ # $ % ^ & * - + =
{ } [ ] : ; ' < > , . ? /
Re-enter new password: # Enter the new password again.
Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any othe
r key for No) : Y
```

b. Enter `Y` to delete the anonymous user account.

```
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 inte
nded only for testing, and to make the installation go a bit smoother. You shou
ld remove them before moving into a production environment.
Remove anonymous users? (Press y|Y for Yes, any other key for No) : Y  # Enter
Y to delete the anonymous user.
Success.
```

c. Enter `Y` to deny remote access by the root account.

```
Disallow root login remotely? (Press y|Y for Yes, any other key for No) : Y # E
nter Y to deny remote access by the root account.
Success.
```

d. Enter `Y` to delete the test database and the access permissions on the database.

```
Remove test database and access to it? (Press y|Y for Yes, any other key for No
) : Y # Enter Y to delete the test database and the access permissions on the d
atabase.
- Dropping test database...
Success.
```

e. Enter `Y` to reload privilege tables.

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y # En
ter Y to reload privilege tables.
Success.
All done!
```

For more information, visit MySQL documentation.

Step 3. Install and configure PHP

1. Installed PHP.
   i. Run the following command to add the IUS repository:

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

   ii. Run the following command to add the Webtatic repository:

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

   iii. Run the following command to install PHP 7 and all required extensions:

```
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 ph
p70w-iconv
```

   iv. Run the following command to check the PHP version:

```
php -v
```

   The following command output indicates that PHP is installed:

```
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
```

2. Configure PHP.
   i. Run the following command to open the PHP configuration file:

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

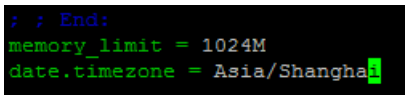
   ii. Move the pointer to the end of the last line. Perform the following operations:
      a. Enter *:$* and press the Enter key to move the pointer to the last line of the file.
      b. Press the *$* key to move the pointer to the end of the line.

iii. Press the /key to enter the edit mode.

iv. Add configurations for the memory limit and time zone at the end of the file.

```
; The maximum memory value allowed for the PHP script. You can increase or decrease
the memory limit.
memory_limit = 1024M
; Set the time zone to Shanghai.
date.timezone = Asia/Shanghai
```

The following figure shows the result.



v. Press the *Esc* key, enter *:wq*, and press the Enter key to save and close the configuration file.

vi. Restart Apache.

```
systemctl restart httpd
```

Step 4. Create a Magento database

1. Run the following command and enter the password of the root user to log on to MySQL:

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

2. Run the following command to create a `magento` database:

```
mysql> CREATE DATABASE magento; # Replace magento with the name of database that you wa
nt to create.
```

3. Run the following commands in sequence to create a user for the `magento` database:

```
mysql> GRANT ALL ON magento.* TO <YourUser>@localhost IDENTIFIED BY '<YourPass>'; # Rep
lace <YourUser> with the account that you want to create and <YourPass> with the passwo
rd that you want to set.
mysql> FLUSH PRIVILEGES;
```

For example, to create an account named `magentoUser` and set its password to `magentoUser1@3`, run the following command:

```
mysql> GRANT ALL ON magento.* TO magentoUser@localhost IDENTIFIED BY 'magentoUser1@3';
mysql> FLUSH PRIVILEGES;
```

4. Enter *exit* and press the Enter key to exit MySQL.

5. (Optional) Check whether the new Magento database and account are available. Perform the following operations:

i. Run the following command to log on to MySQL with the new account and its password:

```
mysql -u <YourUser> -p   # Replace <YourUser> with the account that you created.
```

ii. Run the following command to view the new `magento` database:

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

iii. Run the following command and press the Enter key to exit MySQL:

```
mysql> exit
```

Step 5. Install and configure Composer

Composer is a dependency management tool of PHP. Composer can identify the code repository that is used as the basis for the project. It can also be used to install the depended code repository for the project.

1. Run the following command to install Composer:

```
curl -sS https://getcomposer.org/installer | php
```

2. Run the following command to configure Composer.

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

3. Run the **composer -v** command to check the Composer version.

   The following command output indicates that Composer is installed:

```
  / ____/___  ____ ___  ____  ____  _________  _____
 / /   / __ \/ __ `__ \/ __ \/ __ \/ ___/ _ \/ ___/
/ /___/ /_/ / / / / / / /_/ / /_/ (__  )  __/ /
\____/\____/_/ /_/ /_/ .___/\____/____/\___/_/
                    /_/
Composer version 1.8.5 2019-04-09 17:46:47
```

Step 6. Install and configure Magento

You can install Magento by using different methods and determine whether to install sample data.

- If you install Magento for test purpose only, you can install sample data.
- If you install Magento for production purpose, we recommend that you install Magento and configure it from the start.

In this example, Git is used to download Magento and Composer is used to install Magento.

1. Download Magento.

   i. Run the following command to install Git:

```
yum -y install git
```

ii. Go to the default root directory of the web server.

```
cd /var/www/html/
```

iii. Download Magento.

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

2. (Optional) Run the following command to switch Magento to a stable version:

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

The following output is returned:

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

> **Note** By default, Git downloads and installs the latest Magento version. If you use Magento in a production environment, we recommend that you switch Magento to a stable version. Otherwise, issues may arise when you upgrade and install Magento in the future.

3. Run the following command to move the installation file to the root directory of the web server:

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

> **Note** After you run this command, you can access your Magento website by using `http://<Public IP address of the ECS instance>`. Otherwise, you can access your Magento website only by using `http://<Public IP address of the ECS instance>/magento2`.

4. Run the following commands in sequence to configure appropriate permissions for the Magento file:

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

```
find /var/www/html -type f -print0 | xargs -r0 chmod 640
```

```
find /var/www/html -type d -print0 | xargs -r0 chmod 750
```

```
chmod -R g+w /var/www/html/{pub,var}
```

```
chmod -R g+w /var/www/html/{app/etc,vendor}
```

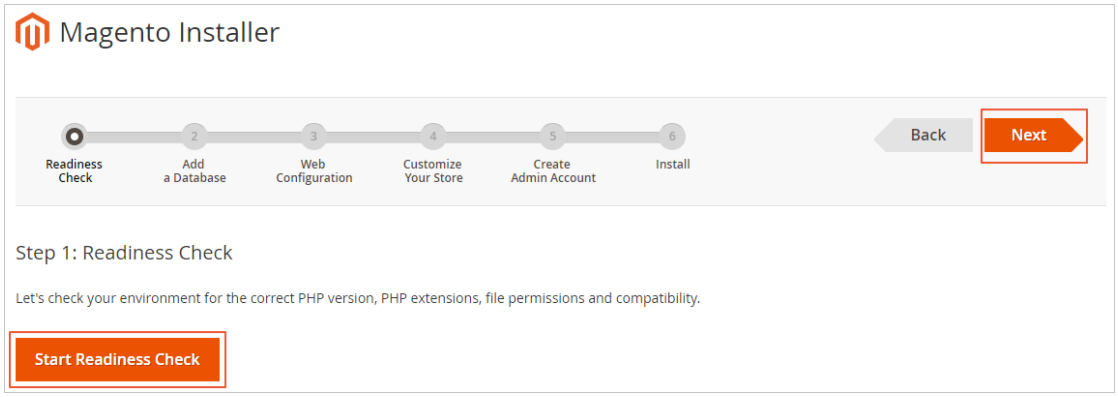
```
chmod 750 /var/www/html/bin/magento
```

5. Run the **composer install** command to install Magento.

Step 7. Configure the client of Magento

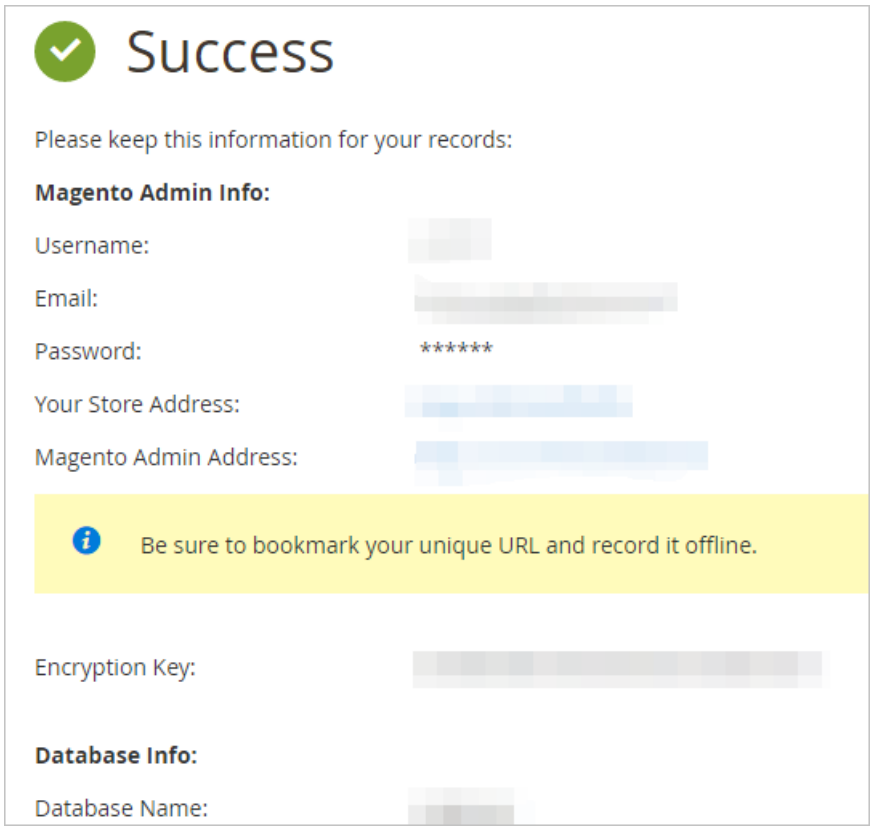
1. Open your browser.
2. In the browser address bar, enter `http://<Public IP address of the ECS instance>`.
   The following page indicates that Magento is installed.

3. Click **Agree and Setup Magento** to start configuring Magento. Perform the following operations:
   i. Check readiness.
      a. Click **Start Readiness Check**.
      b. After the check is complete, click **Next**.



   ii. Add the database.
      a. Enter the account and password of the database that you created. In this example, the user account is `magentoUser` and the password is `magentoUser1@3`.
      b. Enter the name of the created database. In this example, the database name is `magento`.
      c. Click **Next**.
   iii. Complete the settings for web access and click **Next**.
   iv. Fill in the custom store and click **Next**.
   v. Enter the administrator account information and click **Next**.
   vi. Click **Install Now** to install Magento.

The following page indicates that Magento is configured.



Step 8. Add a cron job

Perform the following steps to add a cron job:

1. Run the **crontab -u apache -e** command to configure the jobs to be operated by cron.
2. Press the *i* key to enter the edit mode.
3. Enter the following configurations:

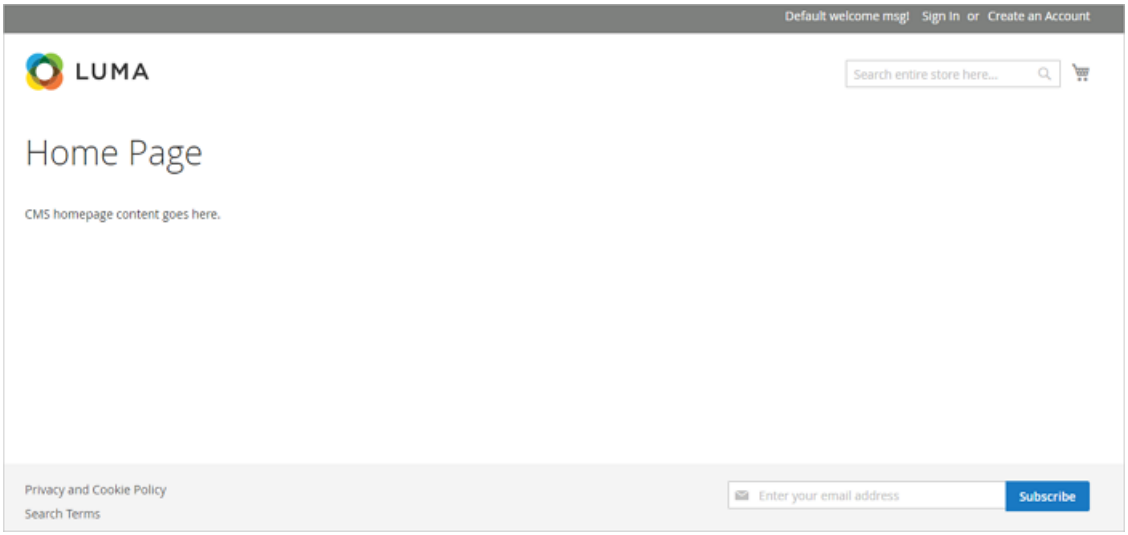
```
*/10 * * * * php -c /etc /var/www/html/bin/magento cron:run
*/10 * * * * php -c /etc /var/www/html/update/cron.php
*/10 * * * * php -c /etc /var/www/html/bin/magento setup:cron:run
```

4. Press the *Esc* key, enter *:wq*, and press the Enter key to save and close the configuration.

For more information about how to use cron jobs in Magento, visit Configure and run cron.

What's next

- Access `http://<Public IP address of the ECS instance>` to go to the following default homepage.



- Access `http://<Public IP address of the ECS instance>/admin` and enter the username and password that you set during the installation. Log on to the management panel to go to the following page.

Related information

- Magento official documentation

5.Build an application

5.1. Build an FTP site on an ECS instance

5.1.1. Manually build an FTP site on a Windows instance

You can build an FTP site on a Windows Elastic Compute Service (ECS) instance so that you can transfer files to or from the instance after you connect to it. This topic describes how to build an FTP site on a Windows instance. This topic is applicable to Windows Server 2008 or later. In the examples, Windows Server 2016 is used.

Prerequisites

One or more Windows instances are created. In the examples, a Windows instance that has the following configurations is used:

- Instance type: ecs.c6.large
- Operating system: Windows Server 2016 64-bit

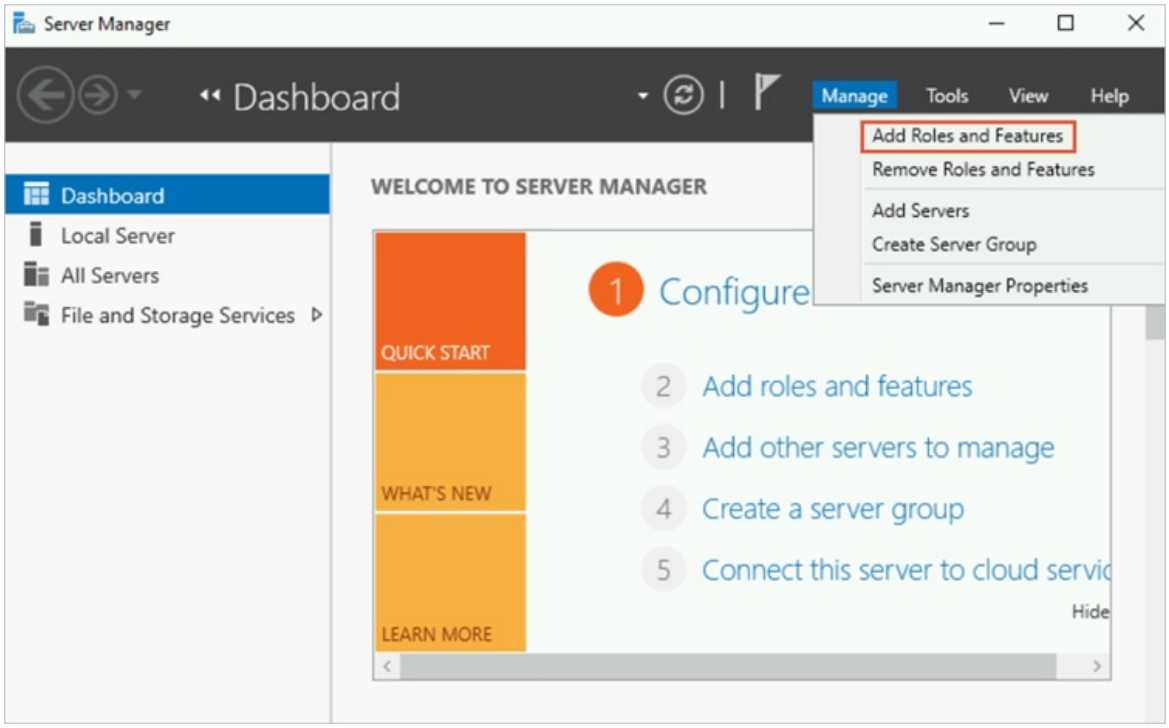
Step 1: Add Internet Information Services (IIS) and FTP server roles

You must install the IIS and FTP services before you can build an FTP site. If you have not installed the IIS and FTP services, perform the following steps to install the services.

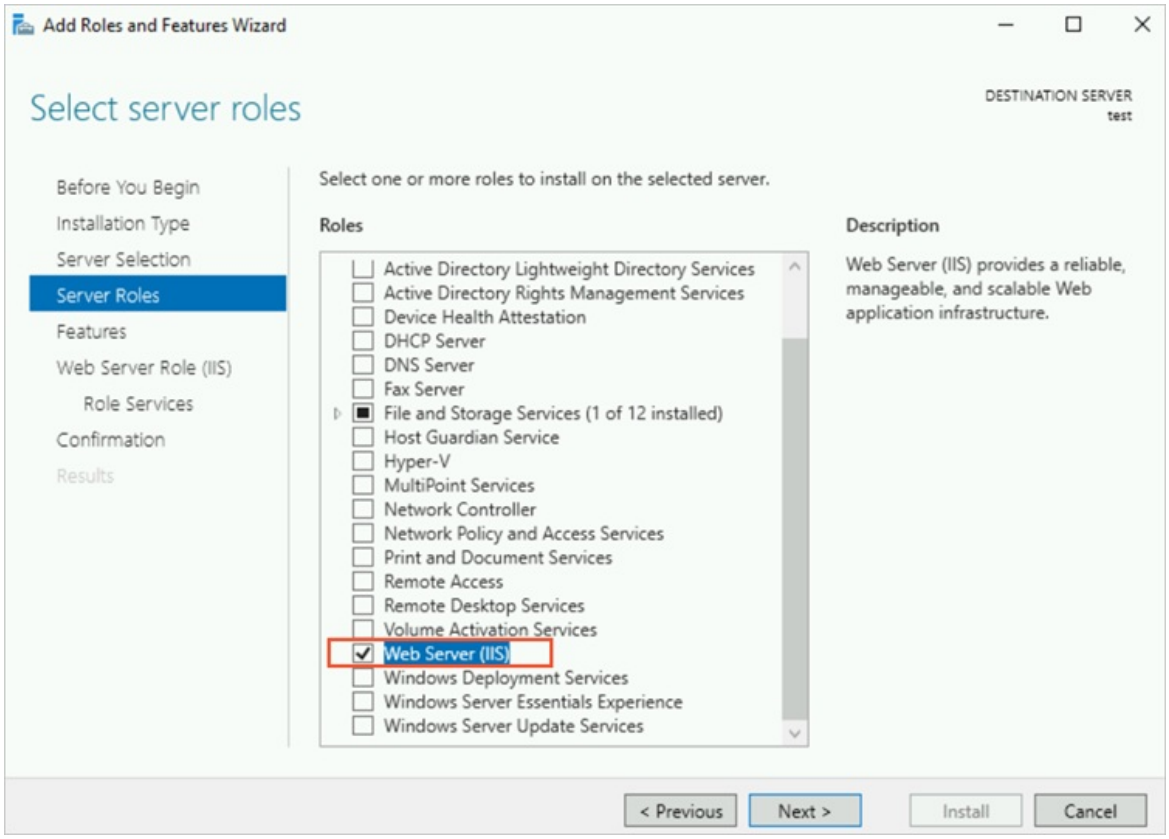
1. Connect to the Windows instance.

   For more information, see Connect to a Windows instance by using a username and password.

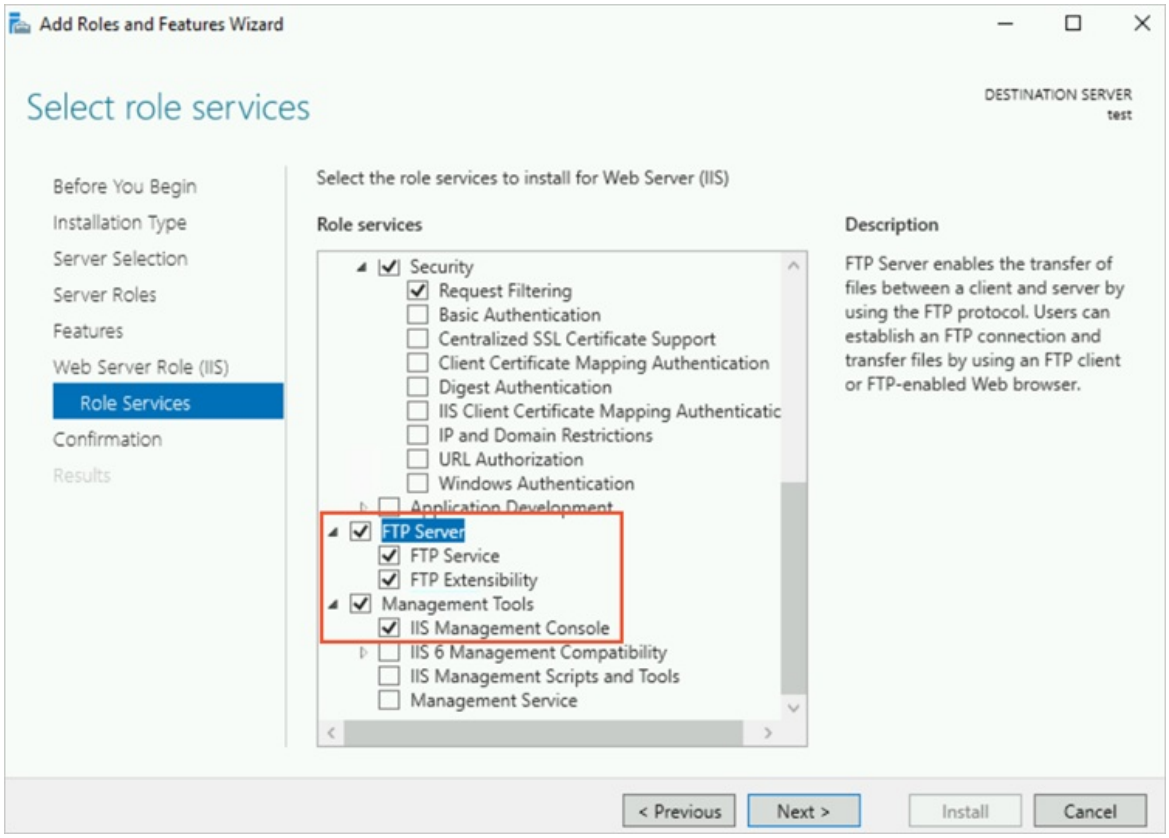
2. In the left-lower corner of the Windows desktop, click the Start ( ) icon. Then, find and click **Server Manager**.
3. In the top navigation bar, choose **Manage > Add Roles and Features**.



4. In the dialog box that appears, accept the default settings and click **Next** until you reach the **Select server roles** step.
5. Select **Web Server (IIS)**. In the dialog box that appears, click **Add Features** and then click **Next**.



6. Accept the default settings and click **Next** until you reach the **Select role services** step.
7. Select **IIS Management Console** and **FTP Server** and click **Next**.

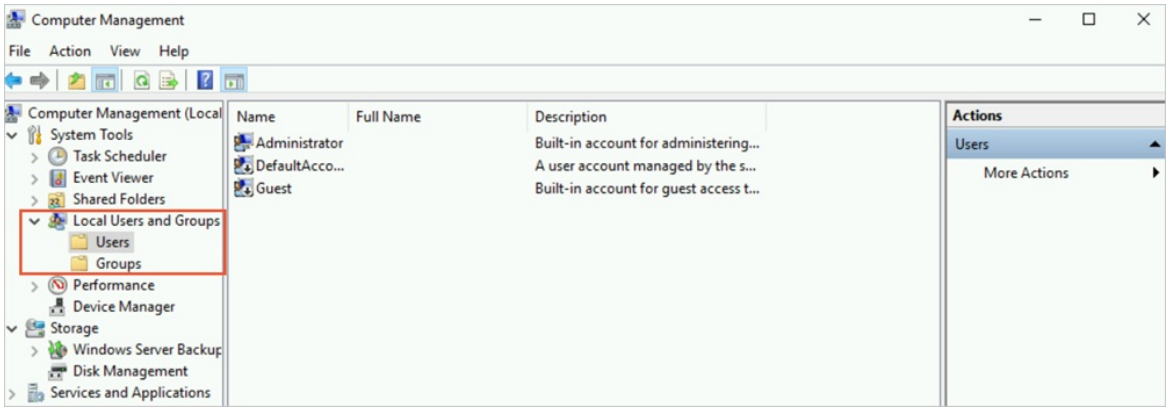


8. Click **Install**. After the IIS and FTP server roles are added, click **Close**.

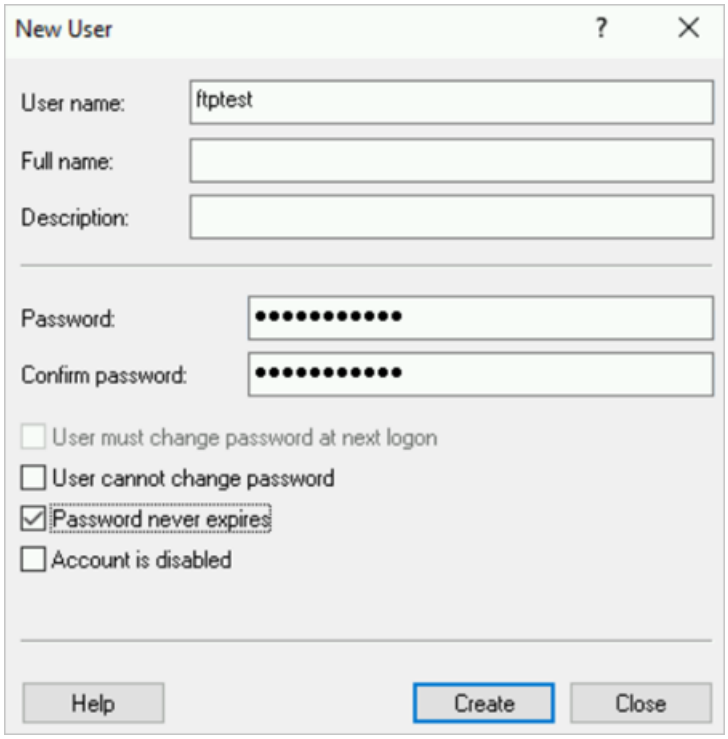
Step 2: Create a Windows user to use to access the FTP site

Create a Windows user to use to access the FTP site to be built. If you want to access the FTP site as an anonymous user with the `anonymous` or `ftp` username, skip the steps described in this section.

1. In the left-lower corner of the Windows desktop, click the Start ( ) icon. Then, find and click **Windows Administrative Tools**.
2. In the **Administrative Tools** window, find and double-click **Computer Management**.
3. In the left-side navigation pane, choose **System Tools > Local Users and Groups > Users**.



4. In the **Actions** column in the right part of the window, click **More Actions** and then click **New User...**



Configure the following parameters or settings:

- User name: In this example, the `ftptest` username is used.
- Password and Confirm password: Enter a password.

  > **Note** The password must contain uppercase letters, lowercase letters, and digits. Keep your password information confidential to prevent data security risks caused by password leaks.

- Password-related settings: Select **Password never expires**.

5. Click **Create** and close the **New User** dialog box.

Step 3: Configure permissions for sharing files

You must create a folder for sharing files with the FTP site and grant the access and modification permissions on the folder. Subsequently, when clients access the FTP site, all files are transferred by using this folder.

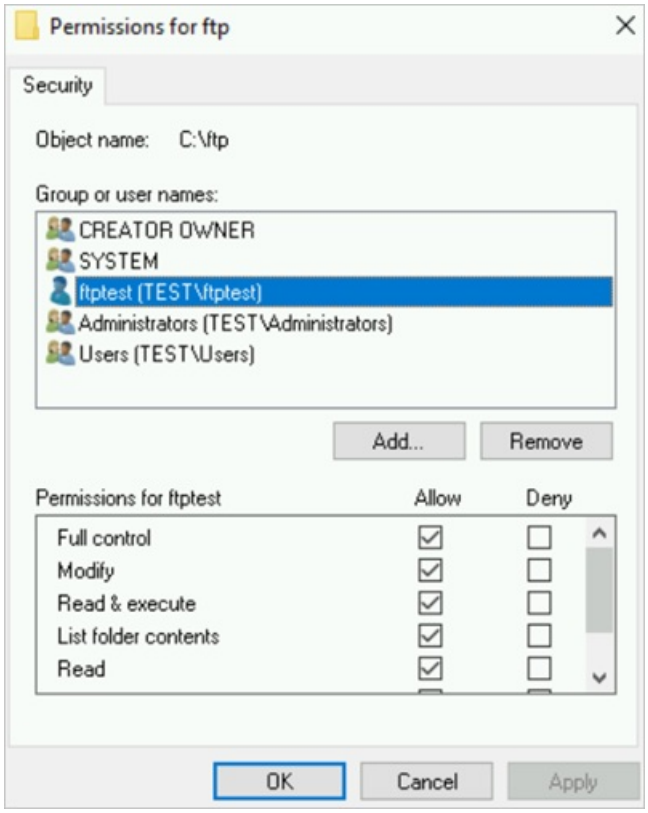
1. On a disk of the Windows instance, create a folder for the FTP site to use.

   In this example, a folder named *ftp* is created on Disk C.

2. Right-click the *ftp* folder and click **Properties**.
3. Click the **Security** tab and then click **Edit**.
4. Click **Add**.
5. In the dialog box that appears, enter *ftptest* as the object name and click **Check Name**.
6. Confirm that the entered object name is correct and click **OK**.
7. In the **Group or user names** section, click the **ftptest** username, configure permissions in the **Permissions for ftptest** section, and then click **OK**.

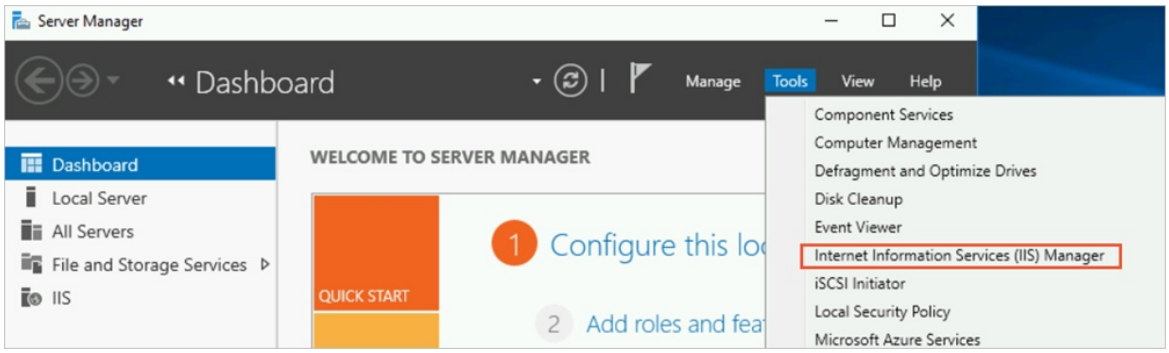
   In this example, all permissions in the Allow column are selected. You can select or clear permissions

in the Allow or Deny column based on your business requirements.



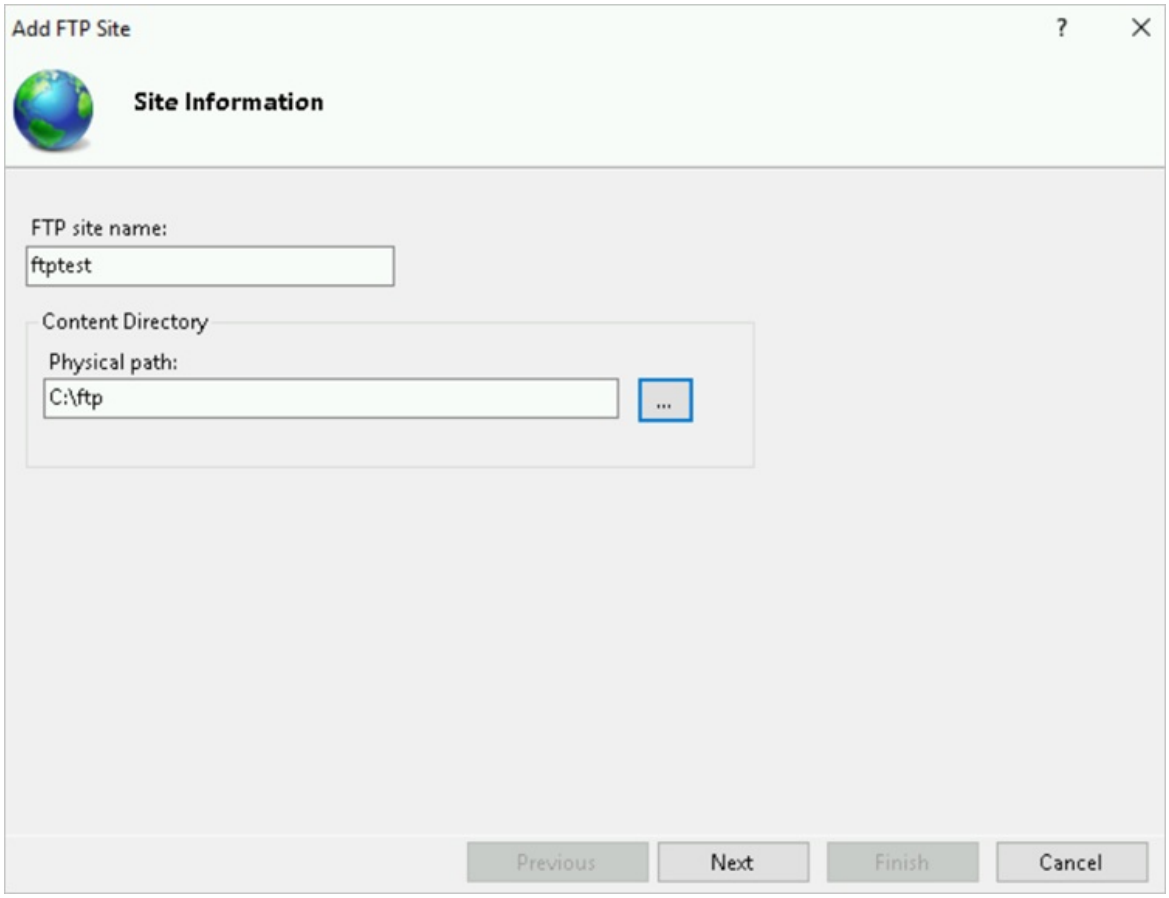
Step 4: Create and configure an FTP site

1. In the left-lower corner of the Windows desktop, click the Start ( ) icon. Then, find and click **Server Manager**.
2. In the top navigation bar, choose **Tools > Internet Information Services (IIS) Manager**.



3. In the left-side navigation pane, choose *<Hostname of the Windows instance>* > **Sites**. Then, in the **Actions** column in the right part, click **Add FTP Site...**

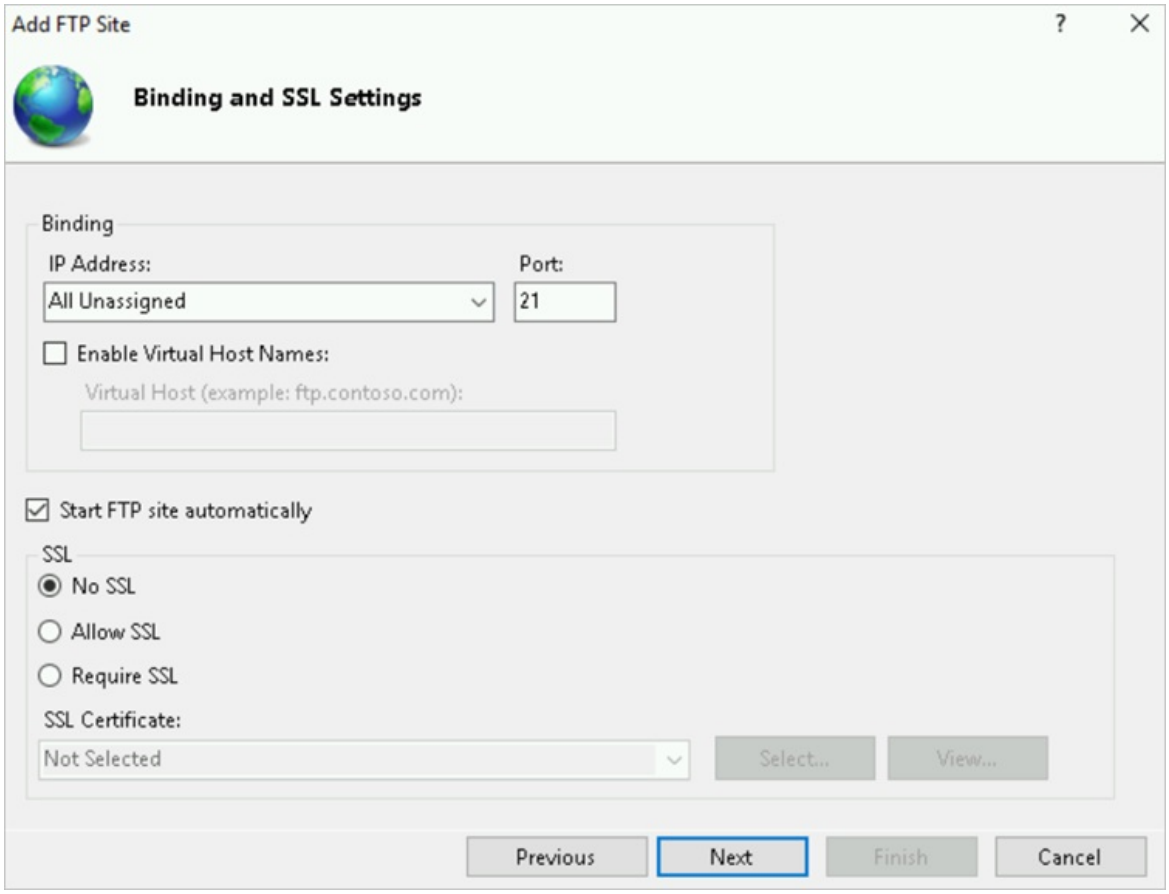
4. In the dialog box that appears, configure parameters and click **Next**.



   Configure the following parameters:

   - FTP site name: Enter a name for the FTP site. Example: `ftptest`.
   - Content Directory: Specify the path to the shared folder required by FTP. In this example, the shared folder is the *ftp* folder created on Disk C.

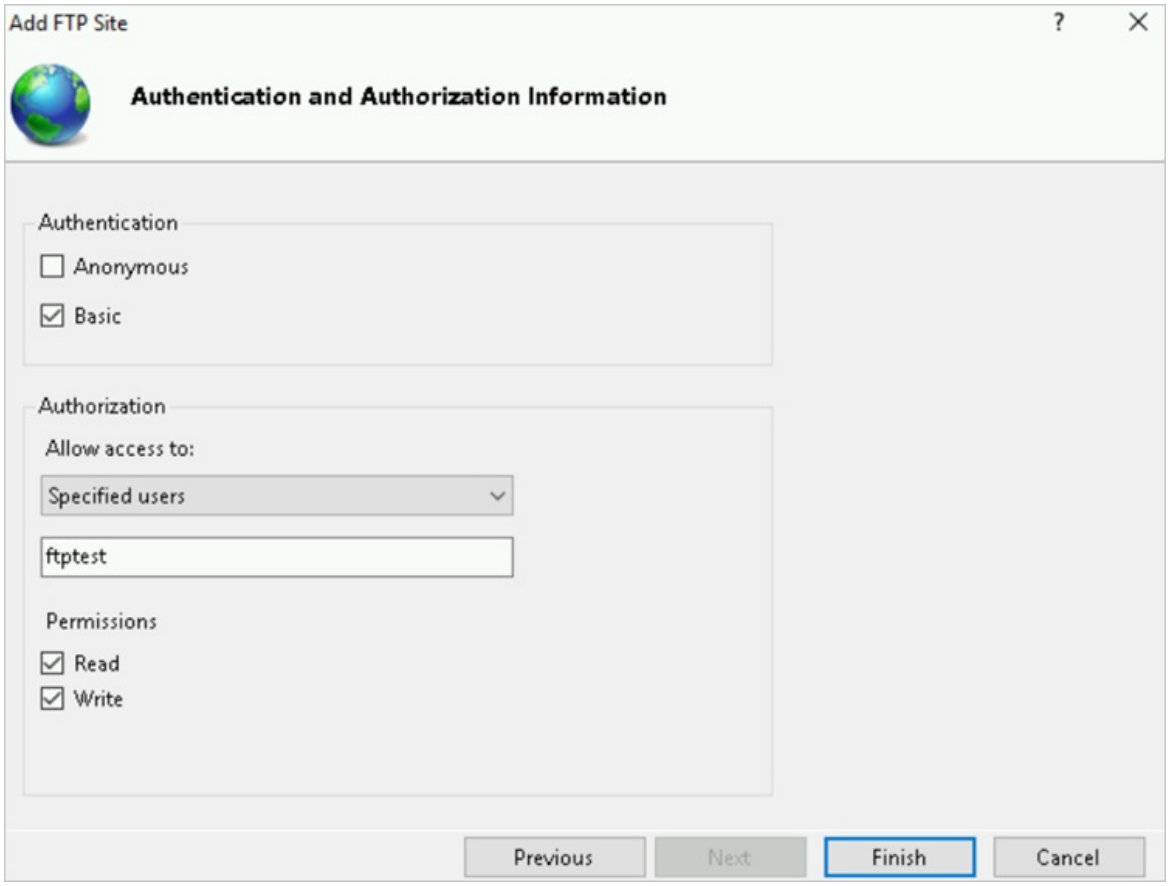
5. Configure the IP address and SSL settings and click **Next**.



Configure the following parameters:

- IP Address: Accept the default settings.
- SSL: In this example, **No SSL** is selected, which indicates that SSL encryption is not required. If you want to secure data transfers and already have an SSL certificate, select **Allow SSL** or **Require SSL**.
  - **No SSL**: SSL encryption is not required.
  - **Allow SSL**: The FTP server is allowed to support both non-SSL and SSL connections with a client.
  - **Require SSL**: SSL encryption is required for communication between the FTP server and a client.
- Accept the default settings for other parameters.

6. Configure authentication and authorization information and click **Finish**.

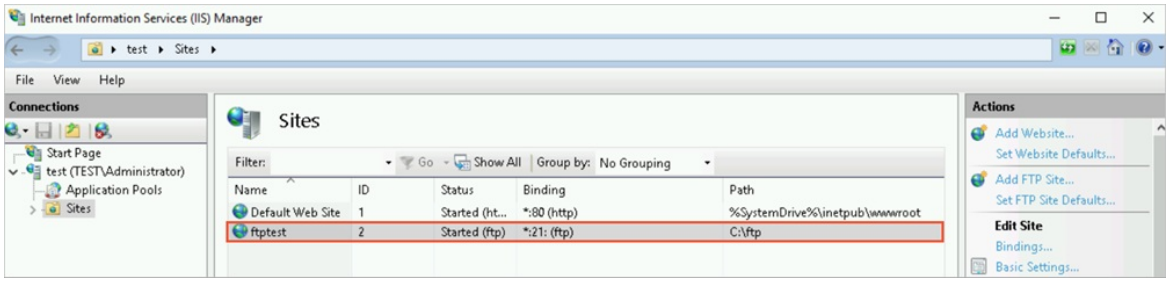


Configure the following parameters:

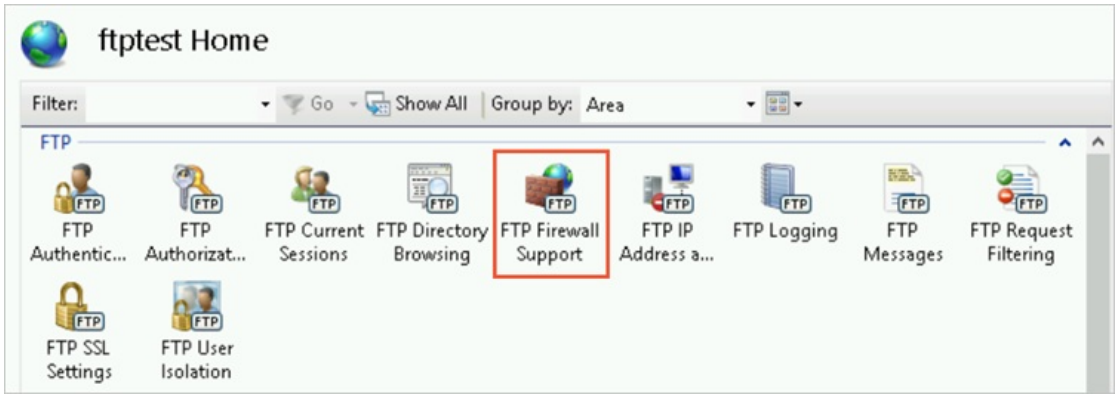
- Authentication: In this example, only **Basic** is selected. Then, you can use the `ftptest` user that you created to access the FTP site. If you do not have security requirements on data transfers, you can select **Anonymous** so that you can access the FTP site as an anonymous user.
  - **Anonymous**: allows users that provides the `anonymous` or `ftp` username to access content.
  - **Basic**: requires users to provide valid usernames and passwords to access content. Basic authentication transmits unencrypted passwords across the network. We recommend that you use basic authentication only when you are certain that the connection between the client and the FTP server is secure, such as when SSL encryption is used.
- Authorization: In this example, Allow access to is set to **Specified users**, and *ftptest* is entered.
  - **All users**: All users are allowed to access the shared folder corresponding to the FTP site.
  - **Anonymous users**: Anonymous users can access the shared folder corresponding to the FTP site.
  - **Specified roles or user groups**: Only specified roles or members of specified groups can access the shared folder corresponding to the FTP site. Enter roles or groups in the corresponding field.
  - **Specified users**: Only specified users can access the shared folder corresponding to the FTP site. Enter usernames in the corresponding field. In this example, ftptest is entered.
- Permissions: Select both **Read** and **Write**.

After the preceding steps are performed, you can view the built FTP site in Internet Information

Services (IIS) Manager.



7. Configure the firewall of the FTP site.
   i. In the Internet Information Services (IIS) Manager window, double-click the FTP site name `ftpt est` to go to the ftptest Home page.
   ii. Double-click **FTP Firewall Support**.



   iii. In the **External IP Address of Firewall** field, enter *<Public IP address of the Windows instance>*.
   iv. In the **Actions** column on the right, click **Apply**. In the message that appears, click **OK**.

Step 5: Configure security groups and the firewall

After you build the FTP site on the Windows instance, you must add inbound rules to the security groups of the instance to allow traffic to port 21 and the passive port range of 1024 to 65535 of the FTP server.

1. In the security groups of the Windows instance, add inbound rules to allow traffic to port 21 and ports in the range of 1024 to 65535.

   For more information, see Add a security group rule.

   > **Note** For more information about security groups, see Security groups for different use casesConfiguration guide for ECS security groups and Typical applications of commonly used ports.

2. (Optional)Configure the firewall of the Windows instance.

   By default, the firewall of the Windows instance is disabled. If your firewall is enabled, allow traffic on TCP port 21 and ports in the range of 1024 to 65535 for the FTP service.

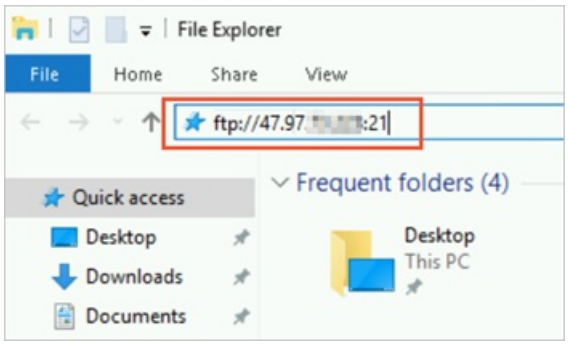
   For more information about firewall settings, see Build an FTP Site on IIS.

Step 6: Use a client to test access to the FTP site

Windows File Explorer, command-line tools, browsers, or third-party FTP connection tools can be used to test the FTP server. In this example, a Windows computer is used as an FTP client and Windows File Explorer is used to access the FTP site.

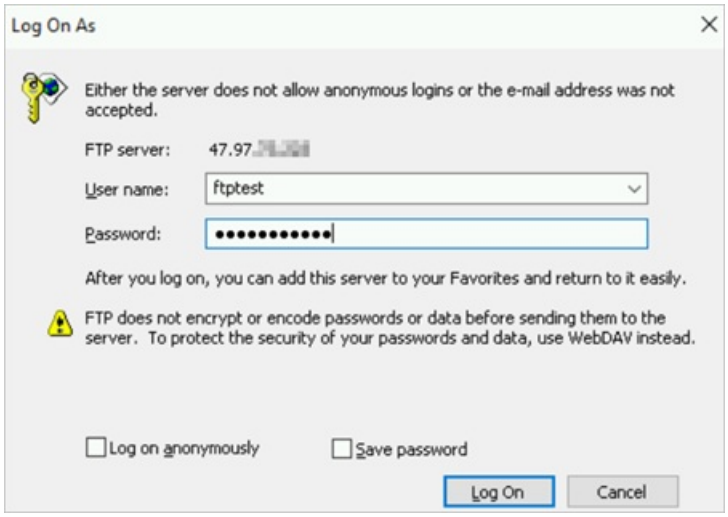
1. On the Windows computer, open Windows File Explorer and enter `ftp://<Public IP address of the FTP site>:21` in the address bar.

   In this example, Window 10 is used.

2. In the **Log On As** dialog box, configure logon credentials and click **OK**.

   In this example, the `ftptest` username and its corresponding password are used as credentials to log on to the FTP site.

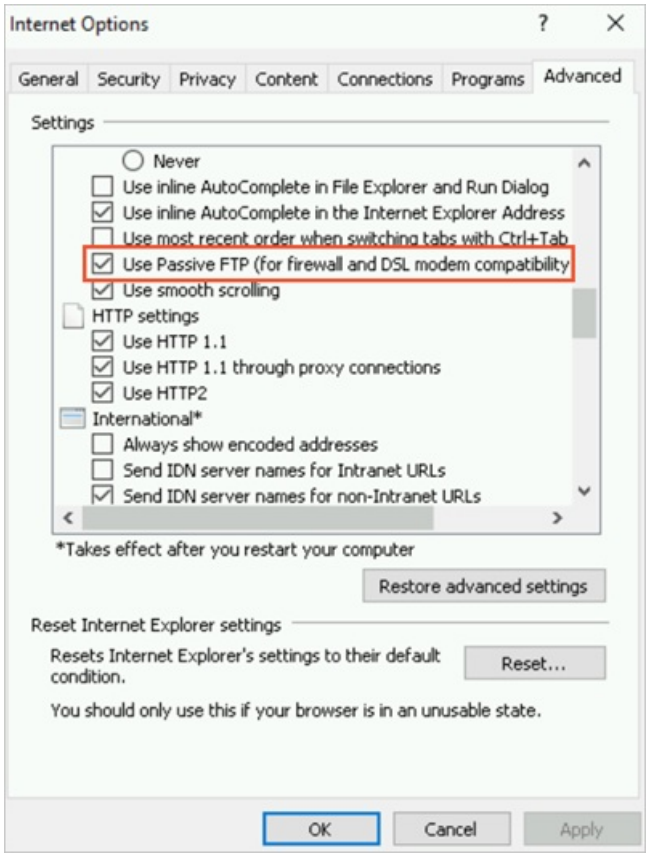
   

   When you use Windows File Explorer to access the FTP site, if **Use Passive FTP** is not enabled for the IE browser on Windows, you cannot access the FTP site. The error codes `200` and `501` are returned. You can perform the following steps to enable **Use Passive FTP** for the IE browser and then access the FTP site again.

   i. In the Windows computer, open the IE browser.

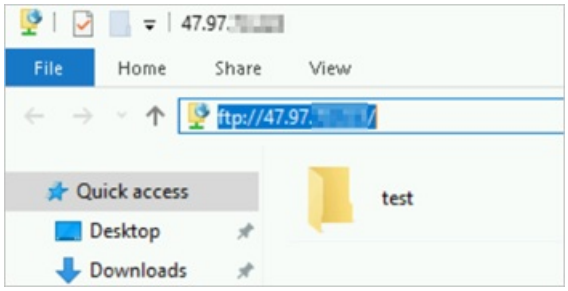
   ii. In the upper-right corner, click the Tools ( ) icon and select **Internet Properties**.

iii. Click the **Advanced** tab. In the Settings section, find and select **Use Passive FTP (for firewall and DSL modem compatibility)**.

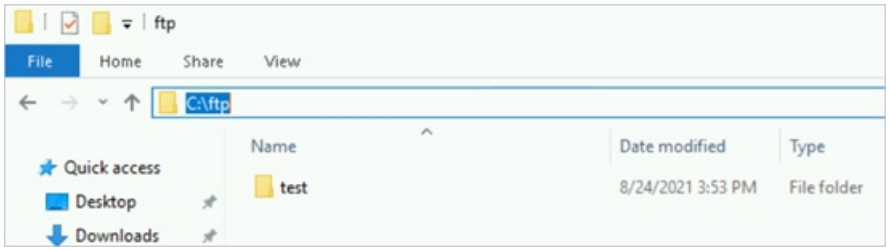


iv. Click **Apply** and then click **OK**.

After you access the shared *ftp* folder that corresponds to the FTP site, you can create a test folder named *test*.



Then, you can log on to the Windows instance again. If the FTP site is built and can be used for data transfers, you can find the *test* folder in the *ftp* folder on Disk C.



What's next

If you want to manage files stored in Object Storage Service (OSS) based on FTP, you can install ossftp. For more information, see Overview.

After ossftp receives a common FTP request, ossftp maps operations on files and folders as operations on OSS.

5.1.2. Manually build an FTP site on a CentOS 8 instance

Very secure FTP daemon (vsftpd) is a lightweight, safe, and easy-to-use FTP server software for Linux. This topic describes how to install and configure vsftpd on a Linux Elastic Compute Service (ECS) instance.

Prerequisites

An ECS instance is created and assigned a public IP address. If no ECS instance is created, create an ECS instance. For more information, see Creation method overview.

Context

FTP is a protocol used to transfer files. FTP is built on a client-server model architecture and supports the following working modes:

- Active mode: The client sends port information to the FTP server, and the server establishes a connection to the port.
- Passive mode: The FTP server enables a port and sends the port information to the client. The client initiates a connection to the port, and the server accepts the connection.

> **Note** Most FTP clients are located in LANs, have no independent public IP addresses, and are protected by firewalls. This makes it difficult for FTP servers in active mode to establish connections to the clients. We recommend that you use passive mode for the FTP server if you do not have special requirements.

FTP supports the following authentication modes:

- Anonymous user mode: In this mode, users can log on to the FTP server without a username or password. This is the least secure authentication mode. In most cases, this mode is used to save unimportant public files. We recommend that you do not use this mode to save files in a production environment
- Local user mode: This authentication mode requires users to have local Linux accounts. This mode is more secure than the anonymous user mode.
- Virtual user mode: Virtual users are dedicated users of the FTP server. Virtual users can access only the FTP service that the Linux system provides. Virtual users cannot access other resources of the system. This way, the security of the FTP server is further enhanced.

In this topic, vsftpd is configured in passive and local user mode. For information about how to configure an FTP server to allow anonymous users to access the FTP server and information about how to use the tools on third-party FTP clients, see FAQ.

The following resources are used in the procedure described in this topic:

- Instance type: ecs.g6.large
- Operating system: CentOS 8.2 64-bit

- vsftpd: 3.0.3

If you use software versions different from the preceding ones, you may need to adjust commands and parameter settings.

Step 1: Install vsftpd

1. Connect to the Linux instance.

   For more information about how to connect to a Linux instance, see Connection methods.

2. Change the CentOS 8 repository address.

3. Run the following command to install vsftpd:

   ```
   dnf install -y vsftpd
   ```

   A command output similar to the following one indicates that vsftpd is installed.

   ```
   Installed:
     vsftpd-3.0.3-31.el8.x86_64


   Complete!
   ```

4. Run the following command to enable the FTP service to automatically start on system startup:

   ```
   systemctl enable vsftpd.service
   ```

5. Run the following command to start the FTP service:

   ```
   systemctl start vsftpd.service
   ```

   Note If the system returns the `Job for vsftpd.service failed because the control process exited with error code` error message when the preceding command is run, check whether the following problems occur and troubleshoot them. If the problems persist, submit a ticket.

   - If the network environment does not support IPv6 addresses, run the **vim /etc/vsftpd/vsftpd.conf** command to change the value of listen_ipv6 from `YES` to `NO`.
   - When the MAC address set for a network interface controller (NIC) in the */etc/sysconfig/network-scripts/ifcfg-xxx* configuration file does not match the actual MAC address of the NIC, run the **ifconfig** command to query the MAC addresses of NICs. Then, add `HWADDR=<Actual MAC address of the NIC>` to the file or change HWADDR in the file to the actual MAC address of the NIC.

6. Run the following command to query the listening port of the FTP service:

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

   A command output similar to the following one indicates that the FTP service is started and listens to port 21.

   ```
   [root@test ~]# netstat -antup | grep ftp
   tcp6       0      0 :::21                   :::*                    LISTEN      47152/vsftpd
   ```

By default, local user mode is enabled. To use the FTP service, you must specify more configurations.

Step 2: Configure vsftpd

In this example, vsftpd is configured in passive and local user mode to ensure data security.

1. Run the following command to create a Linux user for the FTP service. In this example, the ftptest username is used.

   ```
   adduser ftptest
   ```

2. Run the following command. Follow the instructions in the command line to modify the password of the ftptest user.

   ```
   passwd ftptest
   ```

3. Run the following command to create a file directory for the FTP service:

   ```
   mkdir /var/ftp/test
   ```

4. Run the following command to create a test file.

   This test file is used when the FTP client accesses the FTP server.

   ```
   touch /var/ftp/test/testfile.txt
   ```

5. Run the following command to change the owner of the */var/ftp/test* directory to ftptest:

   ```
   chown -R ftptest:ftptest /var/ftp/test
   ```

6. Modify the *vsftpd.conf* configuration file.

   i. Run the following command to open the configuration file of vsftpd.

   If you use the `apt install vsftpd` command to install vsftpd, the path of the configuration file is */etc/vsftpd.conf*.

   ```
   vim /etc/vsftpd/vsftpd.conf
   ```

   ii. Press the /key to enter the edit mode.

   iii. Enable passive mode for the FTP server.

   Configure the following parameters:

   > **Notice** When you modify or add information in the configuration file, take note of the format. For example, an extra space may cause the service to fail to restart.

```
#Use the default values for all parameters except the following parameters:
#Modify the values of the following parameters:
#Disallows anonymous users to log on to the FTP server.
anonymous_enable=NO
#Allows local users to log on to the FTP server.
local_enable=YES
#Listens to IPv4 sockets.
listen=YES
#Add a number sign (#) to the beginning of the row to comment out the following par
ameter:
#Disables listening to IPv6 sockets.
#listen_ipv6=YES
#Add the following parameters at the end of the configuration file:
#Specifies the directory to which local users are directed after they log on.
local_root=/var/ftp/test
#Limits all users to their home directory after they log on.
chroot_local_user=YES
#Uses a list to specify exception users. Exception users are not limited to the hom
e directory after they log on.
chroot_list_enable=YES
#Specifies a file to contain the list of exception users.
chroot_list_file=/etc/vsftpd/chroot_list
#Enables passive mode.
pasv_enable=YES
allow_writeable_chroot=YES
#In this topic, the public IP address of the Linux instance is used.
pasv_address=<The public IP address of the FTP server>
#Specifies the minimum port number of the port range that can be used to transmit d
ata in passive mode.
We recommend that you use ports in a high number range, such as 50000 to 50010. The
se ports provide more secure access to the FTP server.
pasv_min_port=<port number>
#Specifies the maximum port number of the port range that can be used to transmit d
ata in passive mode.
pasv_max_port=<port number>
```

For more information, see vsftpd configuration file and parameters.

iv. Press the *Esc* key to exit the edit mode. Enter *:wq* and press the Enter key to save and close the file.

7. Create the *chroot_list* file, and write the list of exception users to the file.

i. Run the following command to create the *chroot_list* file:

```
vim /etc/vsftpd/chroot_list
```

ii. Press the /key to enter the edit mode.

iii. Enter the list of exception users. Exception users are not limited to the home directory and have access to other directories.

iv. Press the *Esc* key to exit the edit mode. Enter *:wq* and press the Enter key to save and close the file.

> **Notice** If exception users do not exist, you must still create the *chroot_list* file. The file can be empty.

8. Run the following command to restart vsftpd:

```
systemctl restart vsftpd.service
```

Step 3: Configure security groups

After you build the FTP site, add inbound rules for security groups to allow traffic on the following FTP ports. For more information, see Add a security group rule.

> **Note** Most clients are located in LANs and can map their private IP addresses to public IP addresses to communicate with external resources. Therefore, the IP addresses returned by the **ipconfig** or **ifconfig** command may not be the actual public IP addresses of the clients. If you cannot log on to the FTP server from the client, check the public IP address of the client.

In passive mode, you must configure the security group rules to allow traffic on port 21 and on all ports in the port range specified by pasv_min_port and pasv_max_port in the */etc/vsftpd/vsftpd.conf* configuration file. The following table describes the configuration details.

Rule direction	Authorization policy	Protocol type	Port range	Authorized object
Inbound	Allow	Custom TCP	21/21	The public IP addresses in CIDR block notation of all clients that need to access the FTP server. Separate the IP addresses with commas (,). To allow all clients to access the FTP server, specify 0.0.0.0/0 as an authorization object.
Inbound	Allow	Custom TCP	pasv_min_port/pasv_max_port. Example: 50000/50010.	The public IP addresses in CIDR block notation of all clients that need to access the FTP server. Separate the IP addresses with commas (,). To allow all clients to access the FTP server, specify 0.0.0.0/0 as an authorization object.

Step 4: Check whether you can access the FTP server from the client

To check whether FTP servers are accessible, you can use FTP clients, Windows command-line tools, or browsers. In this example, a host that runs Windows Server 2012 R2 64-bit operating system is used to describe how to access an FTP server.

1. On the local host, open **This Computer**.
2. In the address bar, enter `ftp://<The public IP address of the FTP server>:<The FTP port>`. In this example, the following public IP address of the Linux instance is used: `ftp:// 121.43.XX.XX:21`
3. In the **Log on as** dialog box, enter the FTP username and password that you configured, and then click **Logon**.

After you log on, you can view the files under the specified directory in the FTP server, for example, the test file named *testfile.txt*.

vsftpd configuration file and parameters

The following section describes the files under the */etc/vsftpd* directory:

- */etc/vsftpd/vsftpd.conf* is the core configuration file of vsftpd.
- */etc/vsftpd/ftpusers* is the blacklist file. Users specified in this file are not allowed to access the FTP server.
- */etc/vsftpd/user_list* is the whitelist file. Users specified in this file are allowed to access the FTP server.

The following section describes the parameters in the *vsftpd.conf* configuration file.

- The following table describes the parameters for logon control.

Parameter setting	Description
anonymous_enable=YES	Accepts anonymous users.
no_anon_password=YES	Anonymous users do not need a password to log on to the FTP server.
anon_root= (none)	Specifies the home directory of anonymous users.
local_enable=YES	Accepts local users.
local_root= (none)	Specifies the home directory of local users.

- The following table describes the parameters that are used to manage the permissions of users.

Parameter setting	Description
write_enable=YES	Allows all users to upload files.
local_umask=022	Grants local users the permission to upload files.
file_open_mode=0666	Uses umask for permissions to upload files.
anon_upload_enable=NO	Allows anonymous users to upload files.
anon_mkdir_write_enable=NO	Allows anonymous users to create directories.
anon_other_write_enable=NO	Allows anonymous users to modify and delete files.
chown_username=lightwiter	Specifies the ownership of files that are uploaded by anonymous users.

5.1.3. Manually build an FTP site on a CentOS 7 instance

Very secure FTP daemon (vsftpd) is a lightweight, safe, and easy-to-use FTP server software for Linux. This topic describes how to install and configure vsftpd on a Linux Elastic Compute Service (ECS) instance.

Prerequisites

An ECS instance is created and assigned a public IP address. If no ECS instance is created, create an ECS instance. For more information, see Creation method overview.

Context

FTP is a protocol used to transfer files. FTP is built on a client-server model architecture and supports the following working modes:

- Active mode: The client sends port information to the FTP server, and the server establishes a connection to the port.
- Passive mode: The FTP server enables a port and sends the port information to the client. The client initiates a connection to the port, and the server accepts the connection.

> **Note** Most FTP clients are located in LANs, have no independent public IP addresses, and are protected by firewalls. This makes it difficult for FTP servers in active mode to establish connections to the clients. We recommend that you use passive mode for the FTP server if you do not have special requirements.

FTP supports the following authentication modes:

- Anonymous user mode: In this mode, users can log on to the FTP server without a username or password. This is the least secure authentication mode. In most cases, this mode is used to save unimportant public files. We recommend that you do not use this mode to save files in a production environment
- Local user mode: This authentication mode requires users to have local Linux accounts. This mode is more secure than the anonymous user mode.
- Virtual user mode: Virtual users are dedicated users of the FTP server. Virtual users can access only the FTP service that the Linux system provides. Virtual users cannot access other resources of the system. This way, the security of the FTP server is further enhanced.

In this topic, vsftpd is configured in passive and local user mode. For information about how to configure an FTP server to allow anonymous users to access the FTP server and information about how to use the tools on third-party FTP clients, see FAQ.

The following resources are used in the procedure described in this topic:

- Instance type: ecs.c6.large
- Operating system: CentOS 7.2 64-bit
- vsftpd: 3.0.2

The commands and parameters used in this topic may vary based on your resources.

Step 1: Install vsftpd

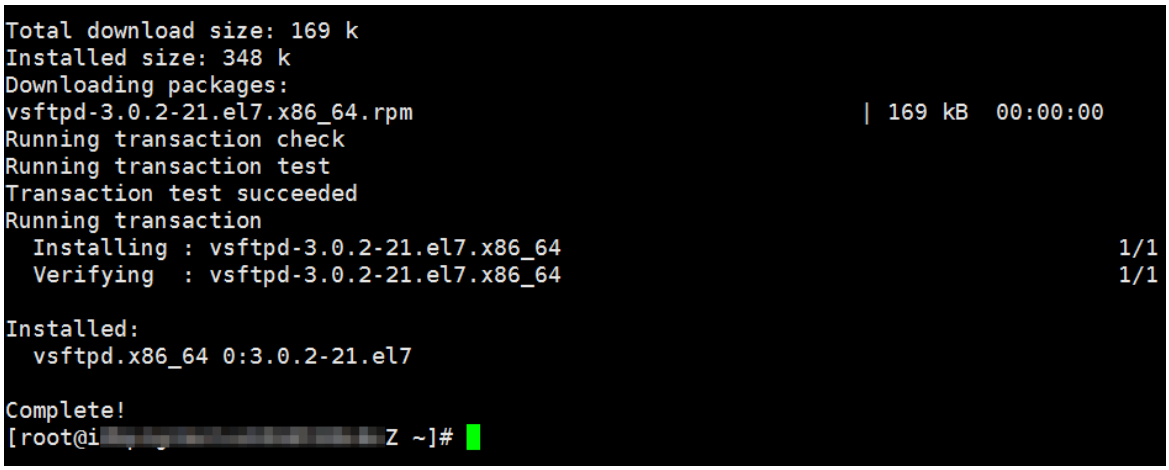
1. Connect to the Linux instance.

   For more information about how to connect to a Linux instance, see Connection methods.

2. Run the following command to install vsftpd:

```
yum install -y vsftpd
```

If the page shown in the following figure appears, vsftpd is installed.



3. Run the following command to enable the FTP service to automatically start on system startup:

```
systemctl enable vsftpd.service
```

4. Run the following command to start the FTP service:

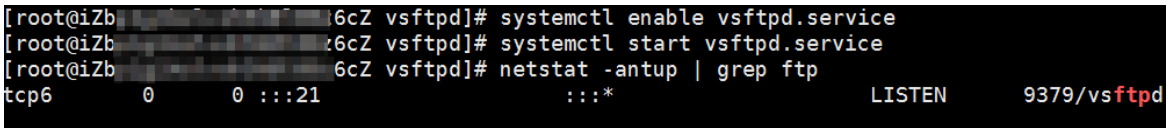
```
systemctl start vsftpd.service
```

> **Note** If the system returns the `Job for vsftpd.service failed because the control process exited with error code` error message when the preceding command is run, check whether the following problems exist and troubleshoot them. If the problems persist, submit a ticket.
>
> - If the network environment does not support IPv6 addresses, run the **vim /etc/vsftpd/vsftpd.conf** command to change the value of listen_ipv6 from `YES` to `NO`.
> - If the MAC address that is specified in the */etc/sysconfig/network-scripts/ifcfg-xxx* configuration file does not match the actual MAC address, run the **ifconfig** command to query the MAC address. Then, add `HWADDR=<The actual MAC address>` to the configuration file. You can also change HWADDR in the configuration file to the actual MAC address.

5. Run the following command to query the listening port of the FTP service:

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

If the following page appears, the FTP service is started and listens to port 21. By default, anonymous access is enabled in vsftpd. You can log on to the FTP server without a username or password. However, you do not have the permissions to modify or upload files.



Step 2: Configure vsftpd

In this example, vsftpd is configured in passive and local user mode to ensure data security.

1. Run the following command to create a Linux user for the FTP service. In this example, the ftptest username is used.

   ```
   adduser ftptest
   ```

2. Run the following command. Follow the instructions in the command line to modify the password of the ftptest user.

   ```
   passwd ftptest
   ```

3. Run the following command to create a file directory for the FTP service:

   ```
   mkdir /var/ftp/test
   ```

4. Run the following command to create a test file.

   This test file is used when the FTP client accesses the FTP server.

   ```
   touch /var/ftp/test/testfile.txt
   ```

5. Run the following command to change the owner of the */var/ftp/test* directory to ftptest:

   ```
   chown -R ftptest:ftptest /var/ftp/test
   ```

6. Modify the *vsftpd.conf* configuration file.

   i. Run the following command to open the configuration file of vsftpd.

   If you use the `apt install vsftpd` command to install vsftpd, the path of the configuration file is */etc/vsftpd.conf*.

   ```
   vim /etc/vsftpd/vsftpd.conf
   ```

   ii. Press the /key to enter the edit mode.

   iii. Enable passive mode for the FTP server.

   Configure the following parameters:

   > **Notice** When you modify or add information in the configuration file, take note of the format. For example, an extra space may cause the service to fail to restart.

```
#Use the default values for all parameters except the following parameters:
#Modify the values of the following parameters:
#Disallows anonymous users to log on to the FTP server.
anonymous_enable=NO
#Allows local users to log on to the FTP server.
local_enable=YES
#Listens to IPv4 sockets.
listen=YES
#Add a number sign (#) to the beginning of the row to comment out the following par
ameter:
#Disables listening to IPv6 sockets.
#listen_ipv6=YES
#Add the following parameters at the end of the configuration file:
#Specifies the directory to which local users are directed after they log on.
local_root=/var/ftp/test
#Limits all users to their home directory after they log on.
chroot_local_user=YES
#Uses a list to specify exception users. Exception users are not limited to the hom
e directory after they log on.
chroot_list_enable=YES
#Specifies a file to contain the list of exception users.
chroot_list_file=/etc/vsftpd/chroot_list
#Enables passive mode.
pasv_enable=YES
allow_writeable_chroot=YES
#In this topic, the public IP address of the Linux instance is used.
pasv_address=<The public IP address of the FTP server>
#Specifies the minimum port number of the port range that can be used to transmit d
ata in passive mode.
We recommend that you use ports in a high number range, such as 50000 to 50010. The
se ports provide more secure access to the FTP server.
pasv_min_port=<port number>
#Specifies the maximum port number of the port range that can be used to transmit d
ata in passive mode.
pasv_max_port=<port number>
```

For more information, see vsftpd configuration file and parameters.

iv. Press the *Esc* key to exit the edit mode. Enter *:wq* and press the Enter key to save and close the file.

7. Create the *chroot_list* file, and write the list of exception users to the file.

i. Run the following command to create the *chroot_list* file:

```
vim /etc/vsftpd/chroot_list
```

ii. Press the /key to enter the edit mode.

iii. Enter the list of exception users. Exception users are not limited to the home directory and have access to other directories.

iv. Press the *Esc* key to exit the edit mode. Enter *:wq* and press the Enter key to save and close the file.

> **Notice** If exception users do not exist, you must still create the *chroot_list* file. The file can be empty.

8. Run the following command to restart vsftpd:

```
systemctl restart vsftpd.service
```

Step 3: Configure security groups

After you build the FTP site, add inbound rules for security groups to allow traffic on the following FTP ports. For more information, see Add a security group rule.

> **Note** Most clients are located in LANs and can map their private IP addresses to public IP addresses to communicate with external resources. Therefore, the IP addresses returned by the **ipconfig** or **ifconfig** command may not be the actual public IP addresses of the clients. If you cannot log on to the FTP server from the client, check the public IP address of the client.

In passive mode, you must configure the security group rules to allow traffic on port 21 and on all ports in the port range specified by pasv_min_port and pasv_max_port in the */etc/vsftpd/vsftpd.conf* configuration file. The following table describes the configuration details.

Rule direction	Authorization policy	Protocol type	Port range	Authorized object
Inbound	Allow	Custom TCP	21/21	The public IP addresses in CIDR block notation of all clients that need to access the FTP server. Separate the IP addresses with commas (,). To allow all clients to access the FTP server, specify 0.0.0.0/0 as an authorization object.
Inbound	Allow	Custom TCP	pasv_min_port/pasv_max_port. Example: 50000/50010.	The public IP addresses in CIDR block notation of all clients that need to access the FTP server. Separate the IP addresses with commas (,). To allow all clients to access the FTP server, specify 0.0.0.0/0 as an authorization object.

Step 4: Check whether you can access the FTP server from the client

To check whether FTP servers are accessible, you can use FTP clients, Windows command-line tools, or browsers. In this example, a host that runs Windows Server 2012 R2 64-bit operating system is used to describe how to access an FTP server.

1. On the local host, open **This Computer**.
2. In the address bar, enter `ftp://<The public IP address of the FTP server>:<The FTP port>`. In this example, the following public IP address of the Linux instance is used: `ftp:// 121.43.XX.XX:21`
3. In the **Log on as** dialog box, enter the FTP username and password that you configured, and then click **Logon**.

After you log on, you can view the files under the specified directory in the FTP server, for example, the test file named *testfile.txt*.

vsftpd configuration file and parameters

The following section describes the files under the */etc/vsftpd* directory:

- */etc/vsftpd/vsftpd.conf* is the core configuration file of vsftpd.
- */etc/vsftpd/ftpusers* is the blacklist file. Users specified in this file are not allowed to access the FTP server.
- */etc/vsftpd/user_list* is the whitelist file. Users specified in this file are allowed to access the FTP server.

The following section describes the parameters in the *vsftpd.conf* configuration file.

- The following table describes the parameters for logon control.

Parameter setting	Description
anonymous_enable=YES	Accepts anonymous users.
no_anon_password=YES	Anonymous users do not need a password to log on to the FTP server.
anon_root= (none)	Specifies the home directory of anonymous users.
local_enable=YES	Accepts local users.
local_root= (none)	Specifies the home directory of local users.

- The following table describes the parameters that are used to manage the permissions of users.

Parameter setting	Description
write_enable=YES	Allows all users to upload files.
local_umask=022	Grants local users the permission to upload files.
file_open_mode=0666	Uses umask for permissions to upload files.
anon_upload_enable=NO	Allows anonymous users to upload files.
anon_mkdir_write_enable=NO	Allows anonymous users to create directories.
anon_other_write_enable=NO	Allows anonymous users to modify and delete files.
chown_username=lightwiter	Specifies the ownership of files that are uploaded by anonymous users.

FAQ

- Question 1: What do I do if I am unable to download files from the FTP server when the local host runs a Windows operating system?

Answer: You must perform the following operations to enable the download permission in Internet Explorer.

i. Open Internet Explorer in your local host.

ii. Click the  icon in the upper-right corner of the browser, and then click **Internet Options**.

iii. At the top of the **Internet Options** dialog box, click the **Security** tab.

iv. In the **Select a zone to view or change security settings.** section, click **Internet**, and then click **Custom level...** in the **Security level for this zone** section.

v. Choose **Download > File Download > Enable**, and then click **OK**.

vi. Click **Apply** and then click **OK**.

- Question 2: What do I do if an error is reported when I use a command-line tool or a browser to connect to an FTP server on Windows?

  Answer: You can manually troubleshoot the problem based on the error message about the FTP server. If the problem is difficult to troubleshoot, we recommend that you use a third-party FTP client connection tool such as FileZilla. You can download FileZilla from FileZilla. In this example, FileZilla is used to connect to an FTP server in anonymous mode.

  > **Note** If the error reported persists when the FTP server is connected to, submit a ticket.

  i. On the FTP server on Linux, install vsftpd.

  For more information, see Step 1: Install vsftpd. If vsftpd is installed, skip this step.

  ii. Configure vsftpd as anonymous mode.

  a. Run the following command to modify the */etc/vsftpd/vsftpd.conf* configuration file.

  If you use the `apt install vsftpd` command to install vsftpd, the path of the configuration file is */etc/vsftpd.conf*.

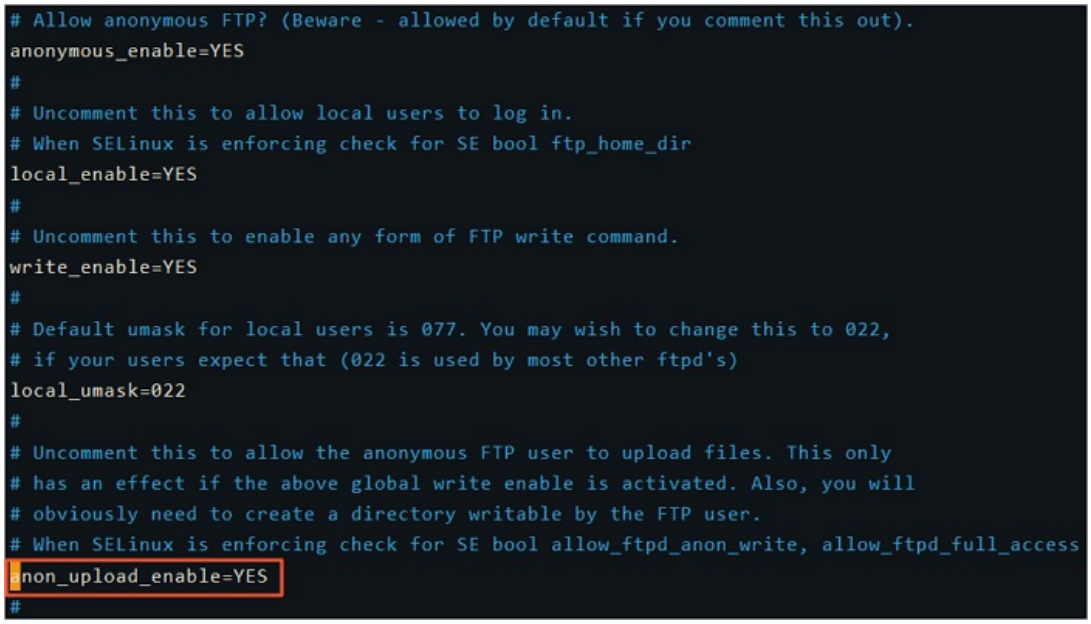
  ```
  vim /etc/vsftpd/vsftpd.conf
  ```

  b. Press the /key to enter the edit mode.

  c. Comment out the permissions and set `anon_upload_enable` to YES to allow anonymous users to upload files.

d. Press the *Esc* key to exit the edit mode. Enter *:wq* and press the Enter key to save and close the file.

The following figure shows the modified configuration file.



e. Run the following command to change the permissions of the */var/ftp/pub* directory and grant write permissions to FTP users:

/var/ftp/pub is the default file directory of the FTP service.

```
chmod o+w /var/ftp/pub/
```

f. Run the following command to reload the configuration file:

```
systemctl restart vsftpd.service
```

iii. Download and install FileZilla.

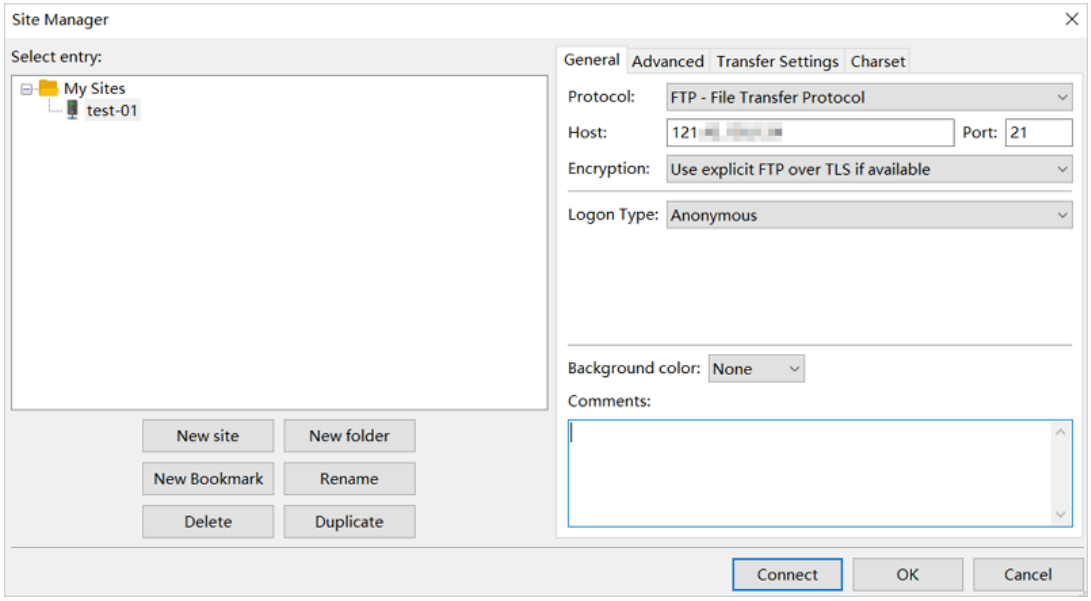
iv. Use FileZilla to connect to the FTP server in anonymous mode.

a. Open the FileZilla client.

b. In the top navigation bar, choose **File > Site Manager**.

c. In the lower-left corner of the **Site Manager** dialog box, click **New site**.

d. Enter a name for the new site and configure the new site.



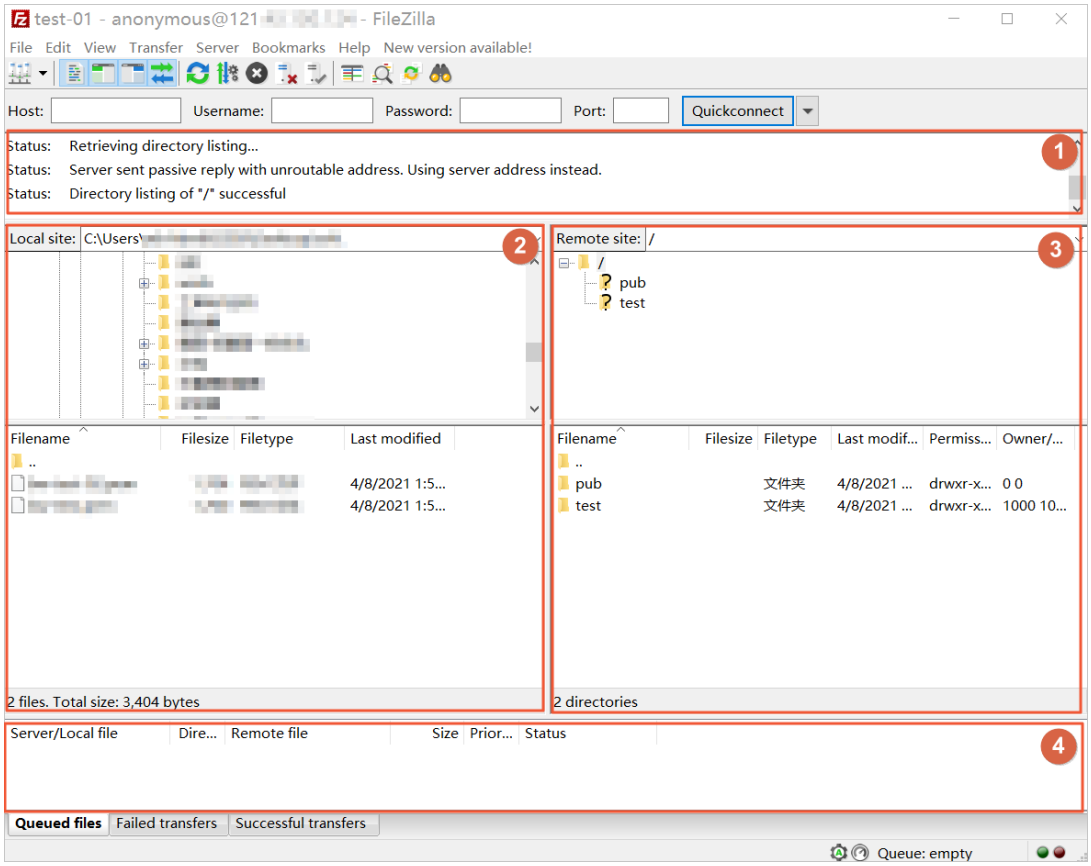
The following list describes the parameters:

- Name: a custom site name. Example, `test-01`.
- Protocol: FTP.
- Host: the public IP address of the FTP server. In this topic, the value is the public IP address of the Linux instance. For example, `121.43.XX.XX`.
- Port: 21.
- Logon Type: Anonymous.

  In this example, an FTP client is used to connect to the FTP server in anonymous mode. If you want to manage access to the FTP server, set the logon type to normal and configure the username and password.

e. Click **Connect**.

After the FTP server is connected to, you can upload, download, and delete files. The FileZilla interface is shown in the following figure.



The following table describes the sections in the preceding interface.

No.	Description
①	Commands, the connection status of the FTP server, and task execution results are shown.
②	The section for the information about the local host, in which the directory information of the local host is shown.
③	The section for the information about the remote server, in which the directory information of the FTP server is shown. In anonymous mode, the default directory is */pub*.
④	The section for records, in which the queues and logs of the FTP task is shown.

5.2. Install and use GitLab

GitLab is a self-managed Git project repository that was developed by Ruby and provides a web interface for access to public or private projects. This topic describes how to install and use GitLab on an Elastic Compute Service (ECS) instance.

Prerequisites

An ECS instance is created and meets the following requirements:

- The instance has at least two vCPUs and 4 GiB of memory. In this example, the following instance type and operating system are used:
  - Instance type: ecs.c6.large
  - Operating system: CentOS 7.2 64-bit

  For more information about how to create an instance, see Create an instance by using the wizard.
- An inbound rule is added to the security group of the instance to allow traffic on port 80. For more information, see Add a security group rule.

Manually deploy a GitLab environment

1. Install the dependency.

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

   > **Note**
   > - In this example, the instance that runs a CentOS 7.2 64-bit operating system is used. If you use an ECS instance that runs CentOS 8, you cannot find the `policycoreutils-python` dependency when you run the preceding command on the instance. This is because the dependency is not included in the software repository of CentOS 8. The absence of this dependency does not affect the deployment of GitLab. You can ignore this issue and proceed to run subsequent commands.
   > - Additionally, CentOS 8 reached its end of life (EOL). If you use instances that run CentOS 8 operating systems, change the CentOS 8 repository address. For more information, see Change CentOS 8 repository addresses.

2. Configure SSH.
   i. Run the following command to start SSH:

      ```
      sudo systemctl start sshd
      ```

   ii. Run the following command to configure SSH to start on instance startup:

      ```
      sudo systemctl enable sshd
      ```

3. Install Postfix to send notification emails.

   ```
   sudo yum install postfix
   ```

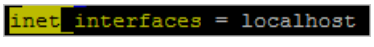
4. Configure Postfix to start on instance startup.

   ```
   sudo systemctl enable postfix
   ```

5. Start Postfix.
   i. Run the following command to open the *main.cf* file:

      ```
      vim /etc/postfix/main.cf
      ```

   ii. Find the code line shown in the following figure and press the /key to enter the edit mode.

iii. Change the code line to `inet_interfaces = all`.

iv. Press the *Esc* key to exit the edit mode, enter `:wq`, and then press the Enter key to save and close the file.

v. Run the following command to start Postfix:

```
sudo systemctl start postfix
```

6. Add the GitLab software package repository.

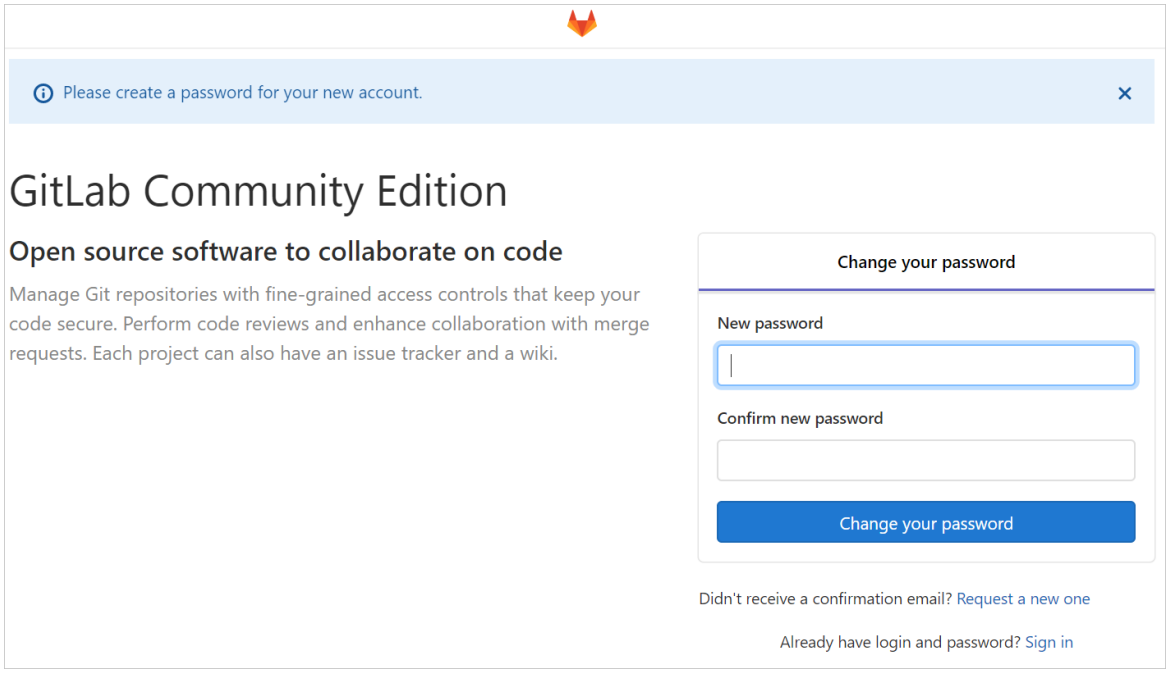
```
curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.rpm.sh | sudo bash
```

7. Install GitLab.

```
sudo EXTERNAL_URL="<Public IP address of the ECS instance>" yum install -y gitlab-ce
```

> **Note** You can choose **Instances & Images > Instances** in the left-side navigation pane in the ECS console and obtain the public IP address of the ECS instance from the Instances page.

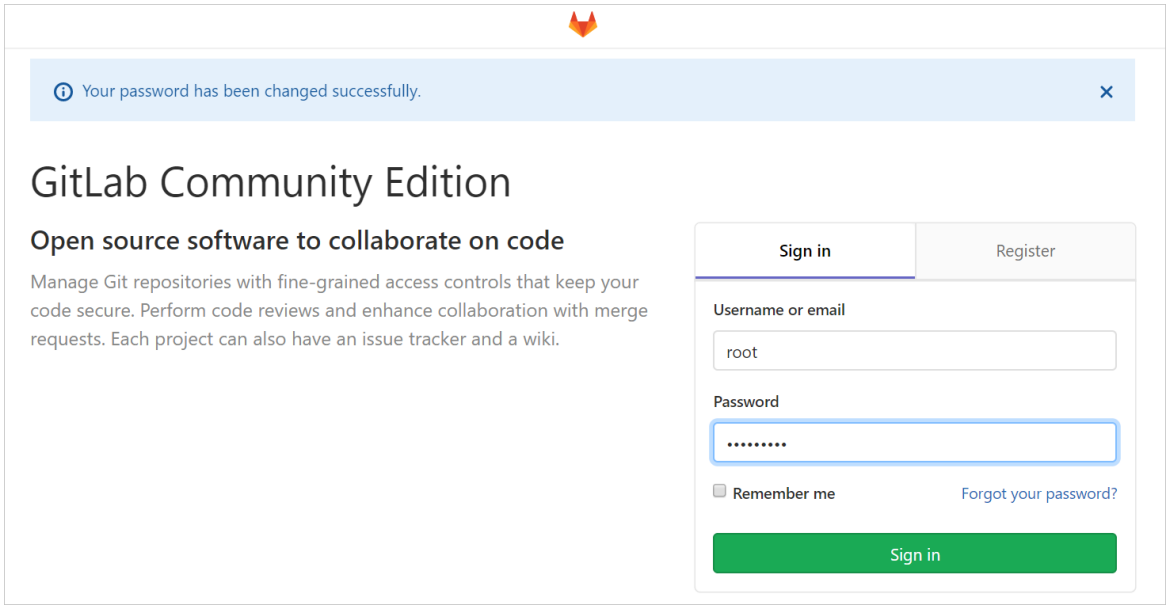
8. Use a browser to access the public IP address of the ECS instance.
The following page indicates that GitLab is installed. You must set the GitLab password.



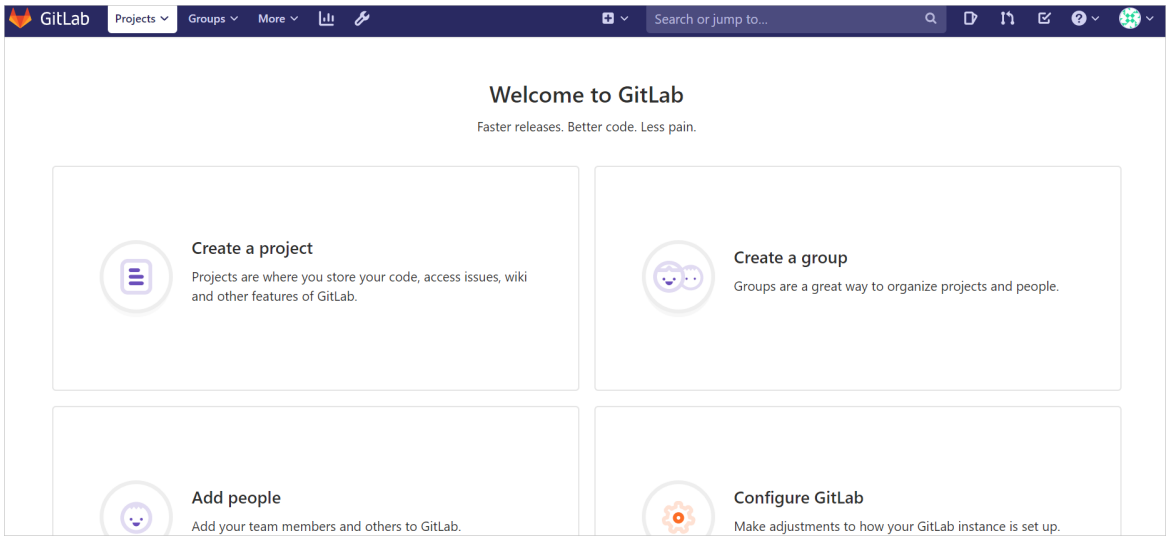
Use GitLab

1. Log on to GitLab.

   In the address bar of your browser, enter the *<Public IP address of the ECS instance>* where GitLab is installed and press the Enter key. The GitLab logon page is displayed. Use the `root` username and the new password that you set at your first logon to log on.

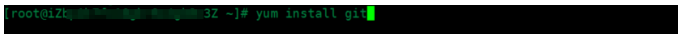


The following page indicates that you are logged on to GitLab.



2. Creates a project.
   i. Use the software repository provided by Linux to install Git.

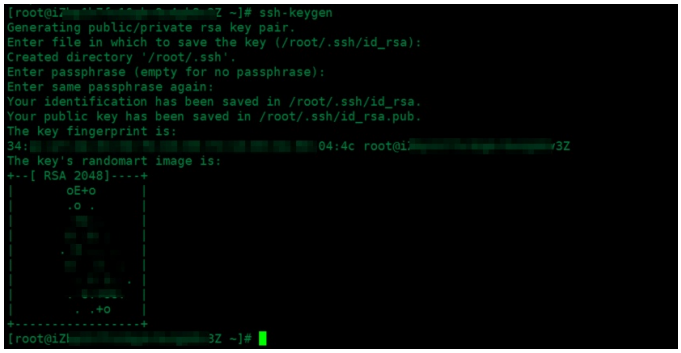
```
yum install git
```



ii. Generate a key file.

Run the following command to generate the *.ssh/id_rsa* key file:

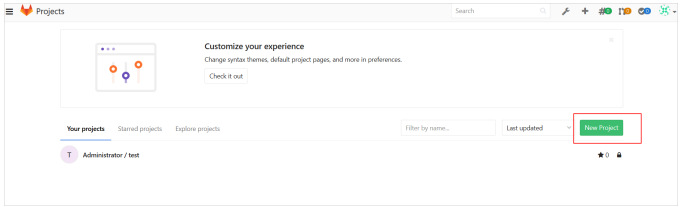
```
ssh-keygen
```



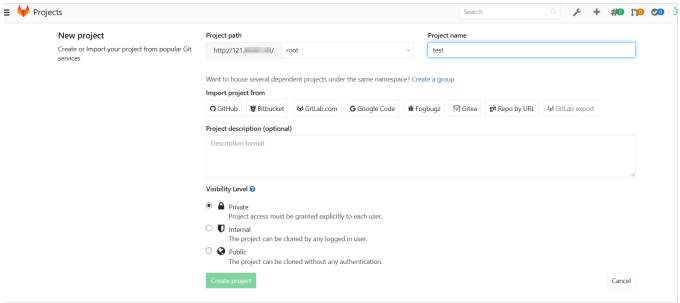
Run the following command to view the content of the *id_rsa.pub* public key file. In the next step, you must paste the content into the SSH key configuration file on the ECS instance where GitLab is installed.

```
cat .ssh/id_rsa.pub
```

iii. On the GitLab homepage, click **New Project** to create a project.

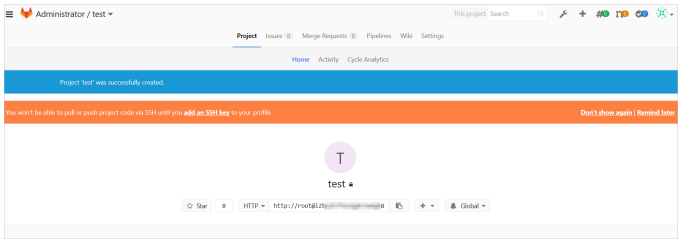


iv. Set the required parameters. Then, click **Create project**.

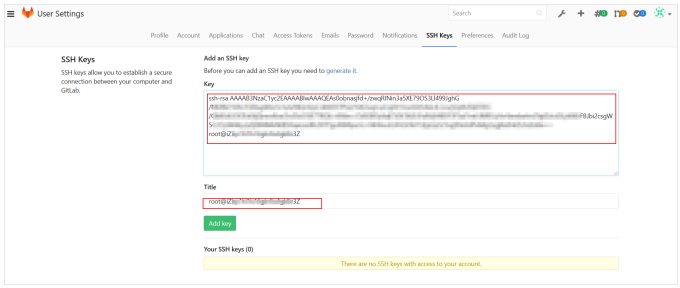


v. Add an SSH key.

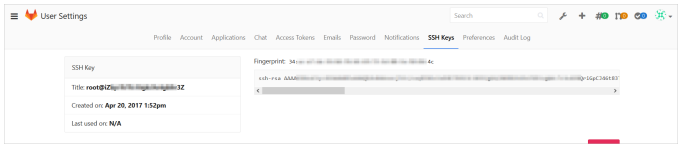
a. Click **add an SSH key**.



b. Import the content of the public key file generated in Step ii.



The following page indicates that the SSH key is added.



vi. Record the address of the new project for future use.

3. Configure GitLab.

i. Add the username of the GitLab user.

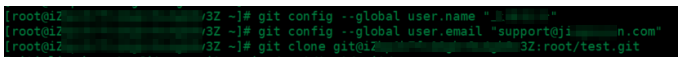
```
git config --global user.name "testname"
```

ii. Add the email address of the GitLab user.

```
git config --global user.email "abc@example.com"
```

iii. Clone the project you created in the previous step to create a local directory that has the same name as the project. All project files are available in the directory.

```
git clone git@iZ****3Z:root/test.git
```



4. Upload a file.

i. Access the local project directory you created in the previous step.

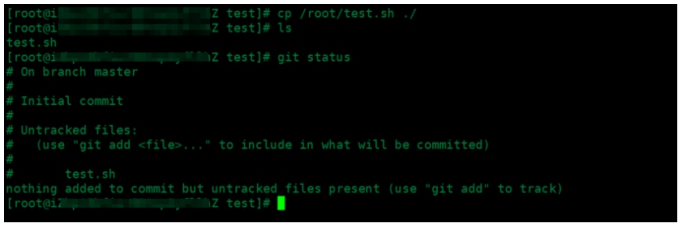
```
cd test/
```

ii. Create a file that you want to upload to GitLab.

```
echo "test" > /root/test.sh
```

iii. Copy the file or directory to the local project directory.

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



iv. Add the *test.sh* file to the index.

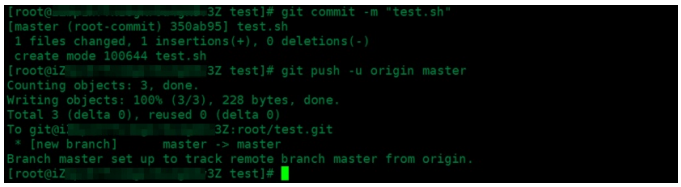
```
git add test.sh
```

v. Commit the *test.sh* file to the local GitLab repository.

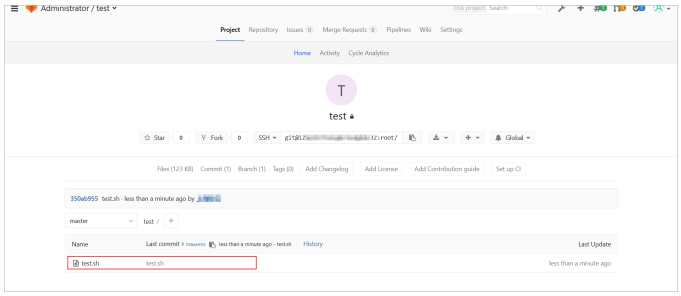
```
git commit -m "test.sh"
```

vi. Push the file to the ECS instance.

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



The *test.sh* file is pushed to GitLab and displayed on the Project page.



5.3. Build Microsoft SharePoint 2016 on an ECS instance

This topic describes how to build Microsoft SharePoint 2016 on an Elastic Compute Service (ECS) instance.

Context

Microsoft SharePoint Portal Server (Microsoft SharePoint) is a portal development environment that allows enterprises to develop intelligent portals. Microsoft SharePoint can be integrated with knowledge bases so that individual users and teams can connect to the environment. Microsoft SharePoint empowers your business by streamlining information processing and providing enterprise-wide service solutions. It allows you to integrate enterprise applications and flexibly choose deployment options and management tools to incorporate information from various systems.

The procedure described in this topic is applicable to users who are familiar with ECS and Windows Server operating systems.

The software described in this topic uses the following versions:

- Operating system: Windows Server 2012 R2 Datacenter
- Database: SQL Server 2014 SP1

The ECS instances described in this topic have the following specifications:

- 4 vCPUs
- 8 GiB of memory

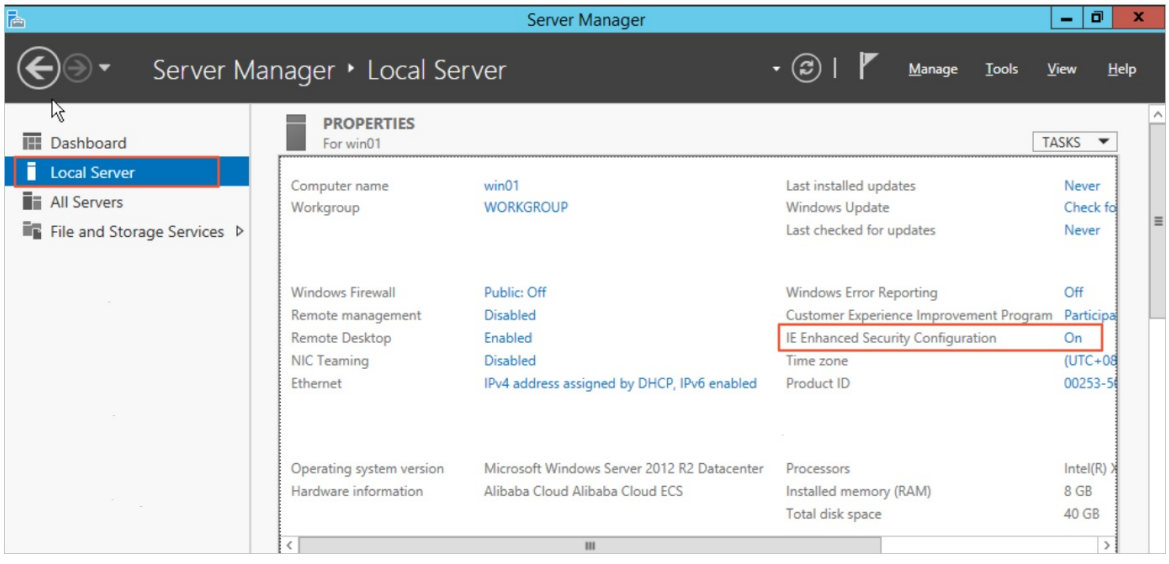
Procedure

To build Microsoft SharePoint 2016 on an ECS instance, perform the following steps:

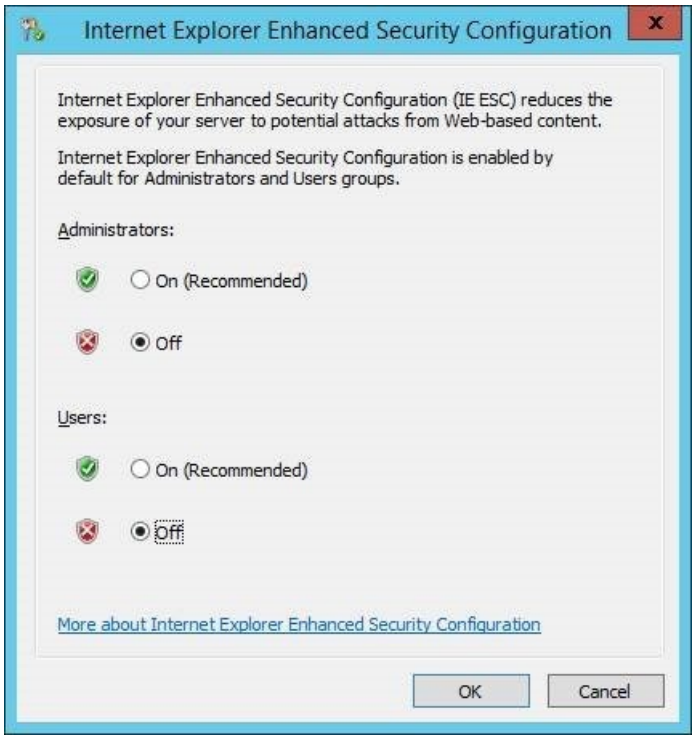
1. Step 1: Add the AD, DHCP, DNS, and IIS services
2. Step 2: Install SQL Server 2014
3. Step 3: Install SharePoint 2016
4. Step 4: Configure SharePoint 2016

Step 1: Add the AD, DHCP, DNS, and IIS services

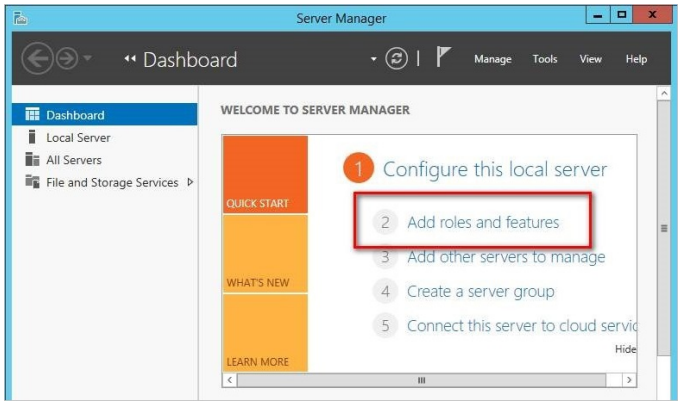
1. Create an ECS instance. For more information, see Create an instance by using the wizard.
2. Connect to the ECS instance.
3. Find and open Server Manager from the Windows taskbar.
4. In the left-side navigation pane, click **Local Server** and find **IE Enhanced Security Configuration** in the PROPERTIES section.



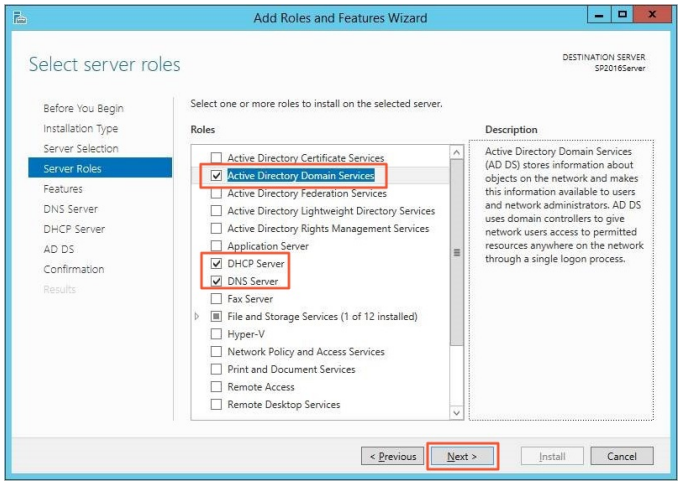
5. Disable Internet Explorer Enhanced Security Configuration.



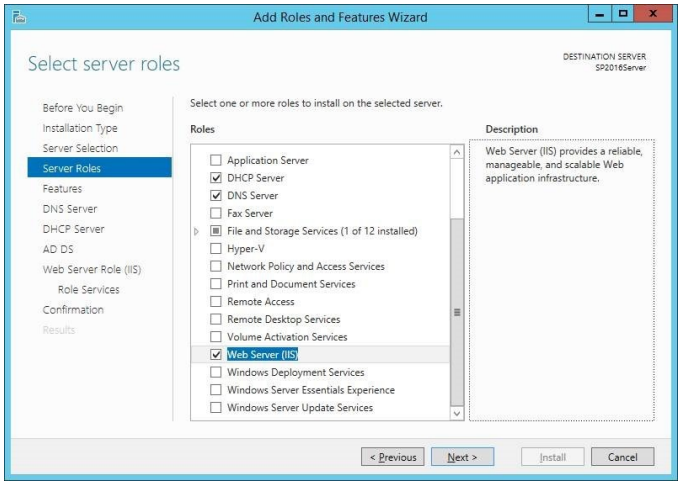
6. Add roles and features including Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), Internet Information Services (IIS), and .NET Framework 3.5.
   i. Click **Add roles and features**.



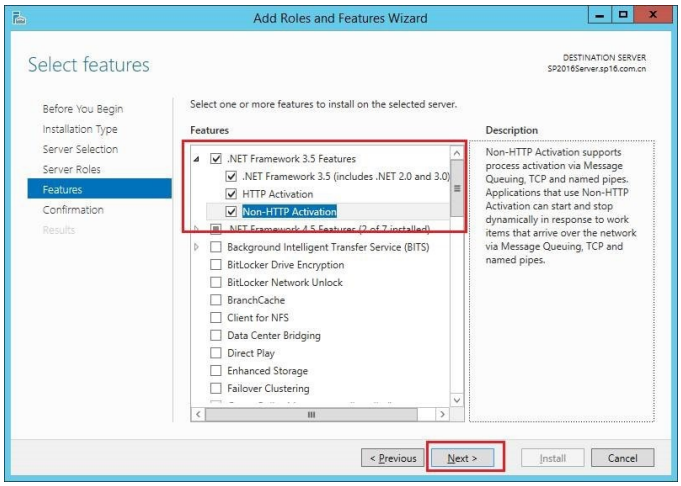
ii. Add the AD, DHCP, and DNS services. Select **Active Directory Domain Services**, **DHCP Server**, and **DNS Server** and click **Next**.



iii. Add the IIS service. Select **Web Server (IIS)** and click **Next**.

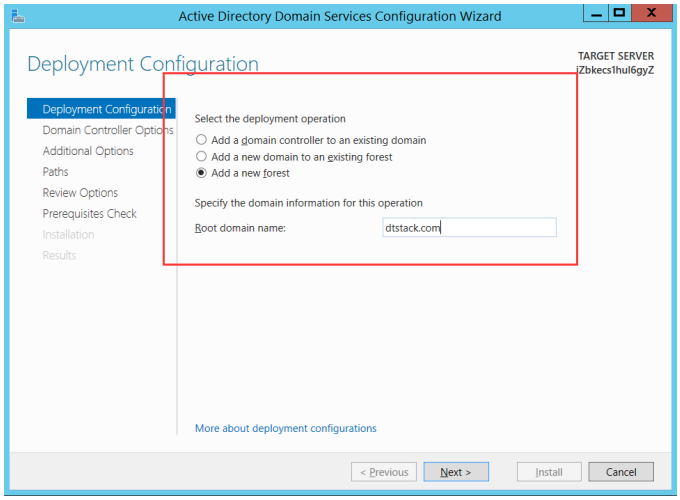


iv. In the **Features** step, select **.NET Framework 3.5 Features**.

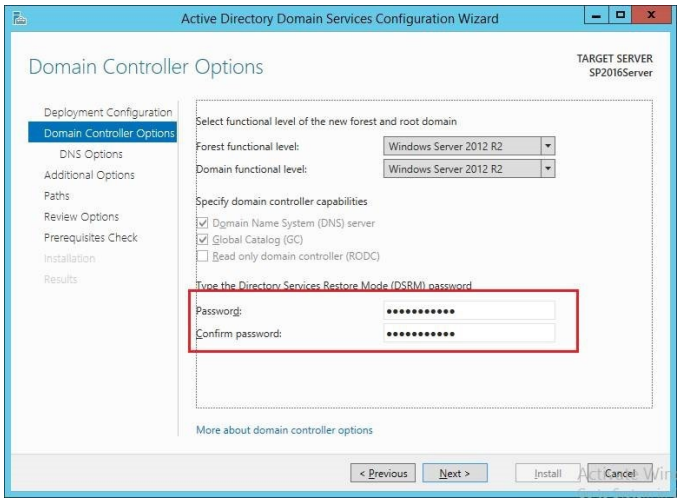


v. Click **Next** to complete Add Roles and Features Wizard.

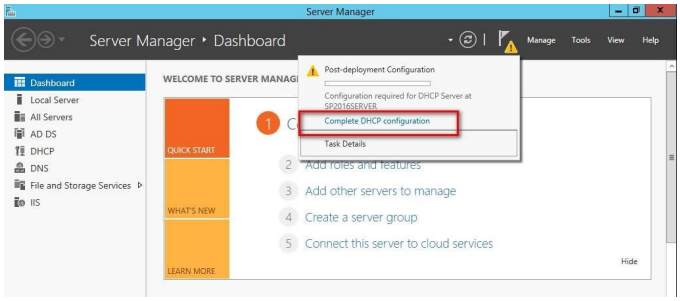
7. Configure the AD service. Select **Add a new forest** and enter a domain name in the **Root domain name** field to create a domain environment.



8. Set a password, confirm the password, and then click **Next** to complete Active Directory Domain Services Configuration Wizard.

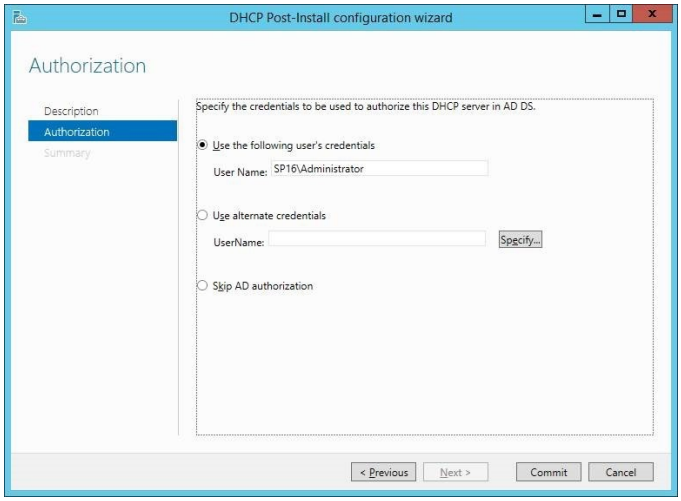


9. Click **Complete DHCP configuration** to configure the DHCP feature.



i. Check the DHCP configuration and click **Next**.

ii. Keep the default configurations and click **Commit** to complete DHCP Post-Install configuration wizard.

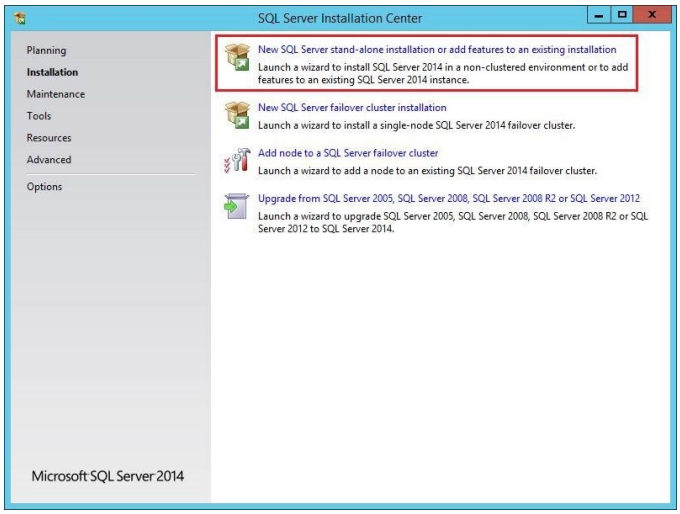


Step 2: Install SQL Server 2014

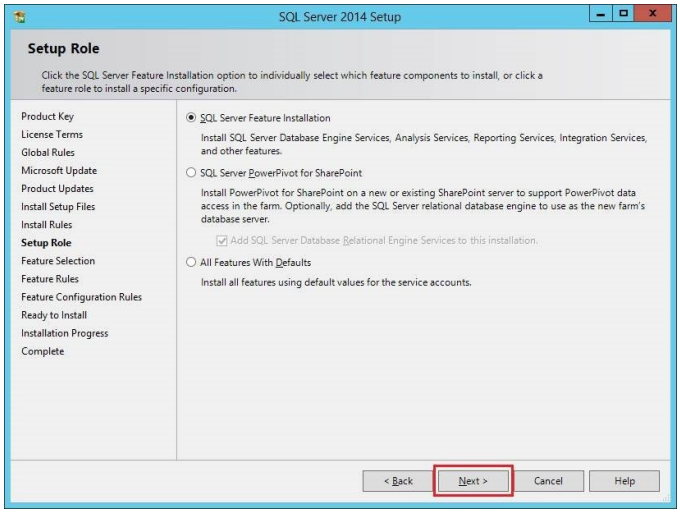
1. You can download SQL Server 2014 from its official website. For more information, visit SQL Server 2014.

   > **Note** You can also download SQL Server 2014 from a third-party website. Make sure that the downloaded software is secure.

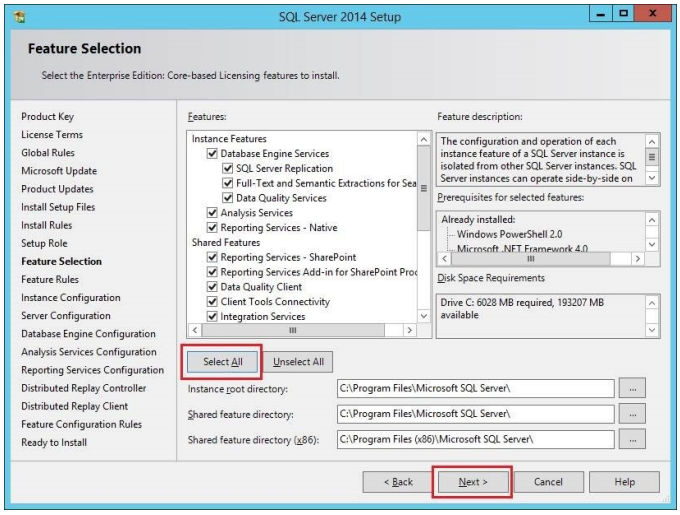
2. Install SQL Server 2014 SP1. Open SQL Server Installation Center and click the first installation option.



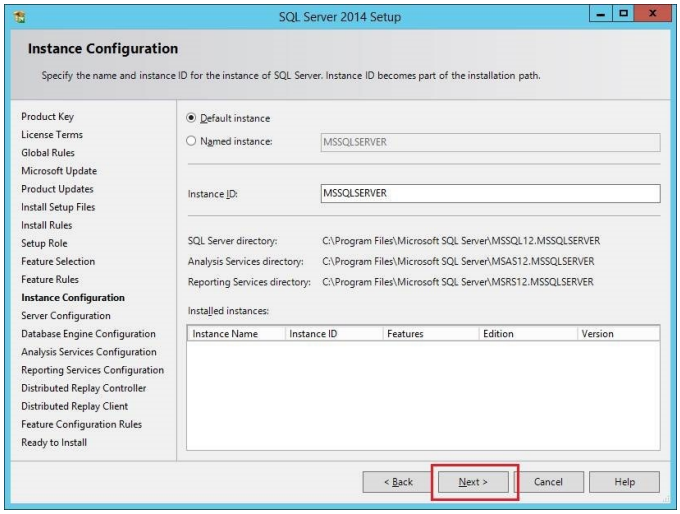
3. Enter the product key and click **Next**.
4. Read and accept the license terms and click **Next**.
5. Complete the installation check and click **Next**.
6. Keep the default configurations and click **Next**.



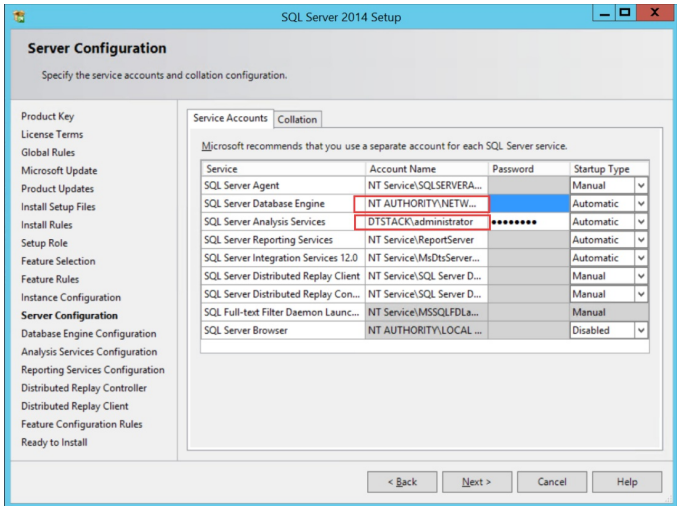
7. Click **Select All** to select all features and click **Next**.



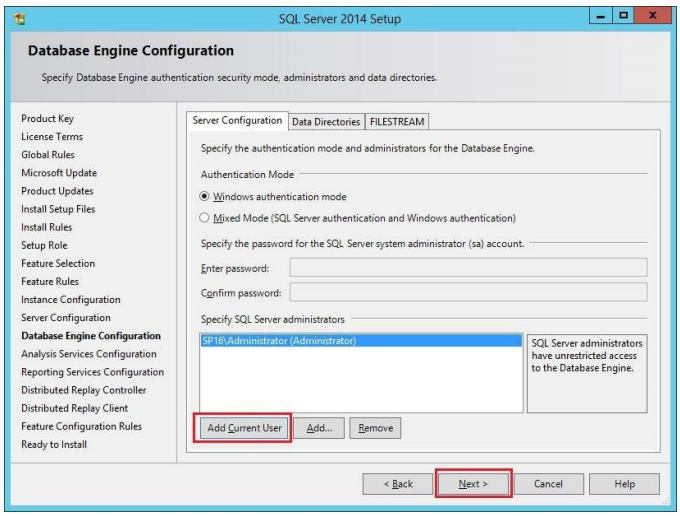
8. Configure the SQL Server instance. Select **Default instance** to use the ID of the instance.



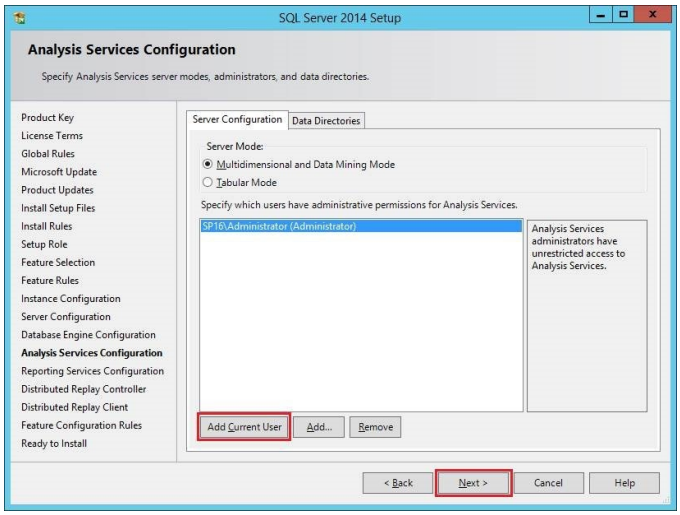
9. Specify the usernames and passwords for **SQL Server Database Engine** and **SQL Server Analysis Services**.



10. Click **Add Current User** to add the current user and then click **Next**.



11. Click **Add Current User** to add the current user and then click **Next**.

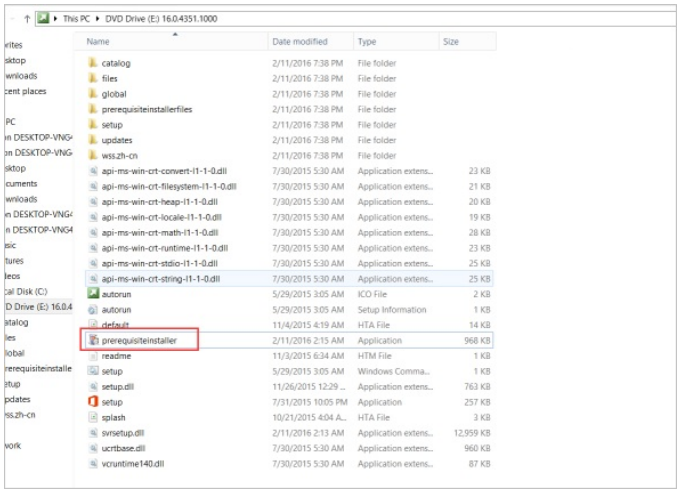


12. Click **Next** to complete the settings of SQL Server 2014.

Step 3: Install SharePoint 2016

1. Install the SharePoint 2016 prerequisite installer. Open the image folder and double-click the

executable file of the prerequisite installer.



2. In the installation wizard, click **Next**.
3. Read and accept the license terms and install necessary components.
4. Open the *Setup.exe* file, enter the product key in the dialog box that appears, read and accept the license terms, and then click **Continue**.
5. Specify the installation directory based on your needs or keep the default configurations and click **Install Now**.
6. After SharePoint 2016 is installed, select **Run the SharePoint Products Configuration Wizard now** and close the wizard.

Step 4: Configure SharePoint 2016

1. Select **Create a new server farm**.
2. Specify configuration database settings and the database access user. The SQL Server 2014 database is installed on your computer. Therefore, you must specify the IP address of your computer in the Database server field.
3. Specify the server role.
4. Select **Specify port number** and enter *10000* in the field. You can also specify another port number based on your needs.
5. Check the configurations and click **Next**.
   After you perform the preceding steps, you can open the SharePoint Central Administration web application.

5.4. Install SharePoint 2016

This topic describes how to install SharePoint 2016.

Prerequisites

-
- An Elastic Compute Service (ECS) instance that runs Windows Server 2012 is created.

Context

To install SharePoint 2016 on an instance, the following environment requirements must be met:

- Basic configurations of the instance:
  - Windows Server 2012
  - CPU: 4 vCPUs. Memory: 8 GB. You can design the architecture and purchase ECS instances based on your actual environments.
- Software environment:
  - SQL server 2012 express
  - SharePoint 2016
  - AD
  - DNS
  - IIS
- Required component: .NET Framework 3.5 for installing SQL Server.

> **Note**
>
> - When you install .NET Framework 3.5, an error may occur in the **Add Roles and Features** step. For more information about how to fix this error, see What do I do if I am unable to install .NET Framework 3.5.1 on Windows Server 2012 R2 or Windows Server 2016 instances?.
> - For more information about the required components of SharePoint, see Microsoft documentation. When you install SharePoint, you are prompted for installing dependency components. If the dependency components cannot be installed, you cannot install SharePoint.

Procedure

1. Connect to the instance. For more information, see Connection methodsGuidelines on instance connection.
2. Build an Active Directory (AD) domain.

   > **Note** You must modify the Security Identifier (SID) before you add a client to a domain. In this topic, only a single ECS instance is used to install SharePoint. Therefore, all roles and features are assigned to the instance. In your actual running environment, do not install SQL Server, AD, and a SharePoint server on the same instance.

3. Install SQL Server 2012 Express.

   Use the default method to install SQL Server. In this topic, the Express edition is used in the test environment. Take note of the following items:

   > **Note**
   >
   > - By default, the Express edition has TCP/IP disabled. You must manually enable TCP/IP.
   > - The Express edition may have no console. You must install an SQL management tool.
   > - We recommend that you use the SQL Server Enterprise edition that provides more features than the Express edition.

4. Install SharePoint 2016.

i. Install the required components of SharePoint.

> **Note** To use the installation wizard, your instance must be authorized to access the Internet. If not, you must manually download the components and run commands to install these components. For more information, see Microsoft documentation.

ii. Restart the instance, run the SharePoint 2016 installation wizard, enter the product key, and then click **Continue**.
Start to install SharePoint 2016.

iii. Run the SharePoint configuration wizard.

iv. Click **Create a new server farm** and click **Next**.

v. Specify configuration database settings and the database access account.

vi. Specify the server role.

vii. Specify the port number for the SharePoint Central Administration web application and configure security settings.

viii. Complete the configurations in the wizard and start to install SharePoint.

ix. Click **Finish**.

After you install SharePoint, you can configure the server farm in the SharePoint Central Administration web application. When you configure the server farm, we recommend that you activate only the required services to prevent unnecessary memory usage.

5.5. Deploy and use Docker

5.5.1. Deploy and use Docker on Alibaba Cloud Linux 3 instances

This topic describes how to deploy and use Docker on an Elastic Compute Service (ECS) instance that runs an Alibaba Cloud Linux 3.2104 64-bit operating system. This topic is intended for developers who are familiar with Linux but new to Alibaba Cloud ECS.

Prerequisites

One or more instances that run an Alibaba Cloud Linux 3.2104 64-bit operating system is created. For more information, see Create an instance by using the wizard.

In this topic, instances that have the following configurations are used:

- Instance type: ecs.g6.large
- Operating system: Alibaba Cloud Linux 3.2104 64-bit
- Network type: Virtual Private Cloud (VPC)
- IP address: public IP address

Context

This topic describes the following operations:

- Deploy Docker. For more information, see the Deploy Docker section.

- Use Docker.
  - For information about how to use Docker, see the Use Docker section.
  - For information about how to create a Docker image, see the Create a Docker image section.

Deploy Docker

1.
2. Run the following command to install Dandified YUM (DNF).

   DNF is the next-generation RPM package manager.

   ```
   yum -y install dnf
   ```

3. Install Docker.

   You can use one of the following methods to install Docker:

   - Install the default Docker (podman-docker) in the DNF repository.
     a. Run the following command to install podman-docker:

        ```
        dnf -y install docker
        ```

     b. Run the following command to check whether Docker is installed:

        ```
        docker images
        ```

        A command output similar to the following one indicates that Docker is installed.

        > **Note** podman-docker installed by using this method has no daemon (systemd). Therefore, you can use Docker without the need to pay attention to the running state of podman-docker in subsequent operations. You do not need to run the **systemctl** command to perform operations.

        ```
        [root@test ~]# docker images
        Emulate Docker CLI using podman. Create /etc/containers/nodocker to quiet msg.
        REPOSITORY  TAG     IMAGE ID  CREATED  SIZE
        ```

   - Install Docker Community Edition (Docker-CE).
     a. Run the following command to add the DNF repository of Docker-CE:

        ```
        dnf config-manager --add-repo=https://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo
        ```

     b. Run the following command to install the DNF repository plug-in that is dedicated for Alibaba Cloud Linux 3:

        ```
        dnf -y install dnf-plugin-releasever-adapter --repo alinux3-plus
        ```

     c. Run the following command to install Docker-CE:

        ```
        dnf -y install docker-ce --nobest
        ```

d. Run the following command to check whether Docker-CE is installed:

```
dnf list docker-ce
```

A command output similar to the following one indicates that Docker-CE is installed.

```
[root@test ~]# dnf list docker-ce
Last metadata expiration check: 0:16:13 ago on Fri 25 Jun 2021 02:10:53 PM CST.
Installed Packages
docker-ce.x86_64                                                    3:20.10.7-3.el8
```

4. Run the following command to start Docker:

```
systemctl start docker
```

5. Run the following command to check the running state of Docker.

```
systemctl status docker
```

A command output similar to the following one indicates that Docker is in the running state.

```
[root@test ~]# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2021-06-25 14:29:34 CST; 2s ago
     Docs: https://docs.docker.com
 Main PID: 14341 (dockerd)
    Tasks: 8
   Memory: 44.5M
   CGroup: /system.slice/docker.service
           └─14341 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

Create a Docker image

1.
2. Create an image.

```
docker build -t webalinux3:v1 .          #Use Dockerfile to create an image. . at the e
nd of the command line specifies the path of Dockerfile and must be provided.
docker images                            #Check whether the image is created.
```

3. Run the container and check its state.

```
docker run -d webalinux3:v1          #Run the container in the background.
docker ps                            #Query the containers that are in the running sta
te.
docker ps -a                         #Query all containers, including those in the sto
pped state.
docker logs CONTAINER ID/IMAGE       #If the container does not appear in the query re
sults, check the startup log to troubleshoot the issue based on the container ID or nam
e.
```

4.
5.

5.5.2. Deploy and use Docker on Alibaba Cloud Linux 2 instances

This topic describes how to deploy and use Docker on an Elastic Compute Service (ECS) instance that runs an Alibaba Cloud Linux 2.1903 LTS 64-bit operating system. This topic is intended for developers who are familiar with Linux but new to Alibaba Cloud ECS.

Prerequisites

One or more instances that run an Alibaba Cloud Linux 2.1903 LTS 64-bit operating system are created. For more information, see Create an instance by using the wizard.

In this topic, instances that have the following configurations are used:

- Instance type: ecs.g6.large
- Operating system: Alibaba Cloud Linux 2.1903 LTS 64-bit

  > Note

- Network type: Virtual Private Cloud (VPC)
- IP address: public IP address

Context

This topic describes the following operations:

- Deploy Docker. For more information, see the Deploy Docker section.
- Use Docker.
  - For information about how to use Docker, see the Use Docker section.
  - For information about how to create a Docker image, see the Create a Docker image section.

Deploy Docker

This section describes how to manually install Docker. You can also purchase a Docker image from Alibaba Cloud Marketplace to deploy Docker on an ECS instance in one click.

1.
2. Run the following command to install Dandified YUM (DNF).

   DNF is the next-generation RPM package manager.

   ```
   yum -y install dnf
   ```

3. Install Docker.

   You can use one of the following methods to install Docker:

   - Install the default Docker from the YUM repository.

     ```
     yum -y install docker
     ```

   - Install Docker Community Edition (Docker-CE).

     a. Run the following command to download the YUM repository of Docker-CE:

     ```
     wget -O /etc/yum.repos.d/docker-ce.repo https://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo
     ```

b. Run the following command to install the YUM repository plug-in that is dedicated for Alibaba Cloud Linux 2.

> **Note** Run this command only if your instance runs an Alibaba Cloud Linux 2 operating system.

```
yum install yum-plugin-releasever-adapter --disablerepo=* --enablerepo=plus
```

c. Run the following command to install Docker-CE:

```
yum -y install docker-ce
```

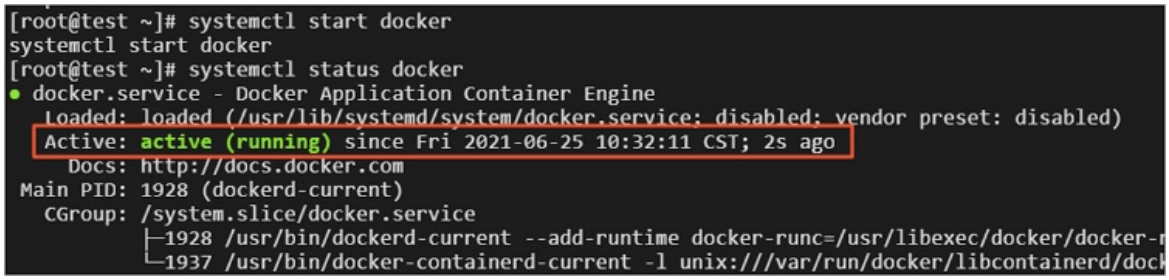
4. Run the following command to start Docker:

```
systemctl start docker
```

5. Run the following command to check the running state of Docker.

```
systemctl status docker
```

If Docker is deployed and in the running state, a command output similar to the following one is returned:



> **Note** To view the version of Docker, run the **docker -v** command.

Use Docker

Docker can be used in the following ways:

- Manage the Docker daemon.

```
systemctl start docker     #Run the Docker daemon.
systemctl stop docker      #Stop the Docker daemon.
systemctl restart docker   #Restart the Docker daemon.
systemctl enable docker    #Configure Docker to run on system startup.
systemctl status docker    #Check the running state of Docker.
```

- Manage images. In the following example, Apache images from Alibaba Cloud Container Registry are used:

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

  - Modify tags. The names of images from Alibaba Cloud Container Registry are long. Use tags to make the images easy to identify.

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

- View existing images.

  ```
  docker images
  ```

- Forcefully delete an image.

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

- Manage containers.
  - Log on to the container. Run the `docker images` command to obtain the ImageId value, which is e1abc****. Then, run the `docker run` command to access the container.

    ```
    docker run -it e1abc**** /bin/bash
    ```

  - Exit the container. Run the `exit` command to exit the container.
  - Add the `-d` parameter to the `run` command to run the container in the background. The `--name` parameter specifies apache as the container name.

    ```
    docker run -d --name apache e1abc****
    ```

  - Access the container that runs in the background.

    ```
    docker exec -it apache /bin/bash
    ```

  - Query the container ID.

    ```
    docker ps
    ```

  - Create an image from the container by using the following command syntax: `docker commit <Container ID or container name> [<Repository name>[:<Tag>]]`.

    ```
    docker commit containerID/containerName repository:tag
    ```

  - For testing and restore purposes, run the image and derive a new image that has a simple name. Then, test the new image.

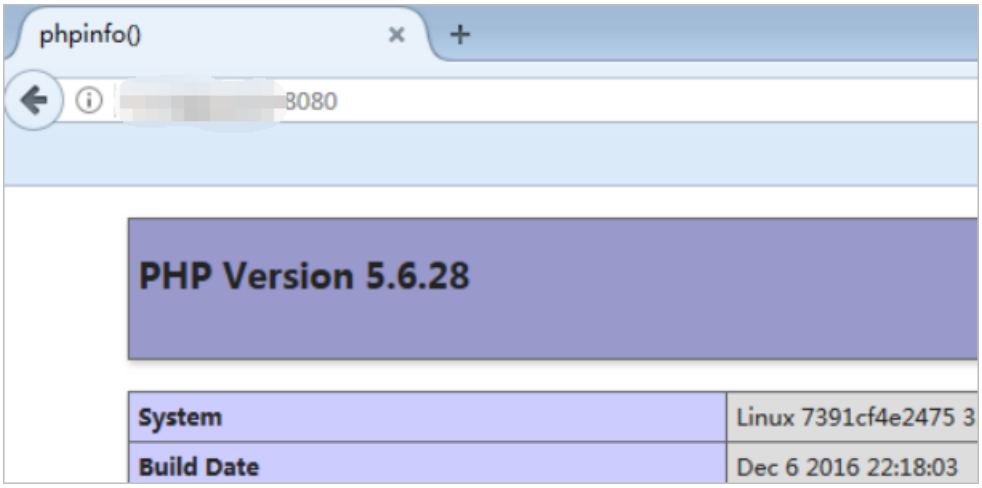
    ```
    docker commit 4c8066cd8**** apachephp:v1
    ```

- Run the container and map port 8080 of the host to the container.

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

In a browser, enter <IP address of the ECS instance>:8080 to connect to the container. A page similar to the one shown in the following figure indicates that the container runs normally.

> **Note** An inbound rule must be added to a security group of the ECS instance to allow inbound traffic on port 8080. For more information, see Add a security group rule.



Create a Docker image

1. Prepare Dockerfile.
   i. Create and edit *Dockerfile*.

```
vim Dockerfile
```

   ii. Press the /key to enter the edit mode. Add the following content to the file:

```
#Declare a base image.
FROM apachephp:v1
#Declare the owner of the base image.
MAINTAINER DTSTACK
#Specify the commands that you want to run before the container starts. You must ap
pend these commands to the end of the RUN command. Dockerfile can contain up to 127
lines. If the total length of your commands exceeds 127 lines, we recommend that yo
u write these commands to a script.
RUN mkdir /dtstact
#Specify the commands that are run on system startup. The last command must be a fr
ontend command that runs constantly. Otherwise, the container exits after all comma
nds are run.
ENTRYPOINT ping www.aliyun.com
```

   iii. Press the *Esc* key. Enter `:wq` and press the *Enter* key to save and exit *Dockerfile*.
2. Create an image.

```
docker build -t webalibabacloudlinux:v1 .    #. at the end of the command specifies the
path of Dockerfile and must be provided.
docker images                                 #Check whether the image is built.
```

3. Run the container and check its state.

```
docker run -d webalibabacloudlinux:v1    #Run the container in the background.
docker ps                                #Query the containers that are in the running
state.
docker ps -a                             #Query all containers, including those in the
stopped state.
docker logs CONTAINER ID/IMAGE           #If the container does not appear in the query
results, check the startup log to troubleshoot the issue based on the container ID or n
ame.
```

4. Create an image.

```
docker commit fb2844b6**** dtstackweb:v1     #Append the container ID and the name and
version number of the new image to the end of the commit parameter.
docker images                                #Query images that are downloaded and crea
ted on premises.
```

5. Push the image to a remote repository.

   By default, the image is pushed to Docker Hub. You must log on to Docker, add a tag to the image, and then name the image in the `<Docker username>/<Image name>:<Tag>` format. Then, the image is pushed to the remote repository.

```
docker login --username=dtstack_plus registry.cn-shanghai.aliyuncs.com #Enter the passw
ord of the image repository after you run this command.
docker tag [ImageId] registry.cn-shanghai.aliyuncs.com/dtstack123/test:[Tag]
docker push registry.cn-shanghai.aliyuncs.com/dtstack123/test:[Tag]
```

5.5.3. Deploy and use Docker on CentOS 8 instances

This topic describes how to deploy and use Docker on an Elastic Compute Service (ECS) instance that runs a CentOS 8.1 64-bit operating system. This topic is intended for developers who are familiar with Linux but new to Alibaba Cloud ECS.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

In this topic, an instance that has the following configurations is used:

- Instance type: ecs.g6.large
- Operating system: CentOS 8.1 64-bit
- Network type: Virtual Private Cloud (VPC)
- IP address: public IP address

Context

This topic describes the following operations:

- Deploy Docker. For more information, see the Deploy Docker section.
- Use Docker.
  - For information about how to use Docker, see the section.
  - For information about how to create a Docker image, see the Create a Docker image section.

Deploy Docker

This section describes how to manually install Docker. You can also purchase a Docker image from Alibaba Cloud Marketplace to deploy Docker on an ECS instance in one click.

1.
2. Change the CentOS 8 repository address.
3. Run the following command to install the dependency of the Docker store driver:

   ```
   dnf install -y device-mapper-persistent-data lvm2
   ```

4. Run the following command to add a stable Docker repository:

   ```
   dnf config-manager --add-repo=https://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo
   ```

5. Run the following command to view the added Docker repository:

   ```
   dnf list docker-ce
   ```

   Sample command output:

   ```
   docker-ce.x86_64        3:19.03.13-3.el7        docker-ce-stable
   ```

6. Run the following command to install Docker:

   ```
   dnf install -y docker-ce --nobest
   ```

7. Run the following command to start Docker:

   ```
   systemctl start docker
   ```

Use Docker

Docker can be used in the following ways:

- Manage the Docker daemon.

  ```
  systemctl start docker     #Run the Docker daemon.
  systemctl stop docker      #Stop the Docker daemon.
  systemctl restart docker   #Restart the Docker daemon.
  systemctl enable docker    #Configure Docker to run on system startup.
  systemctl status docker    #Check the running state of Docker.
  ```

- Manage images. In the following example, Apache images from Alibaba Cloud Container Registry are used:

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

- Modify tags. The names of images from Alibaba Cloud Container Registry are long. Use tags to make the images easy to identify.

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

- View existing images.

  ```
  docker images
  ```

- Forcefully delete an image.

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

- Manage containers.
  - Log on to the container. Run the `docker images` command to obtain the ImageId value, which is e1abc****. Then, run the `docker run` command to access the container.

    ```
    docker run -it e1abc**** /bin/bash
    ```

  - Exit the container. Run the `exit` command to exit the container.
  - Add the `-d` parameter to the `run` command to run the container in the background. The `--name` parameter specifies apache as the container name.

    ```
    docker run -d --name apache e1abc****
    ```

  - Access the container that runs in the background.

    ```
    docker exec -it apache /bin/bash
    ```

  - Query the container ID.

    ```
    docker ps
    ```

  - Create an image from the container by using the following command syntax: `docker commit <Container ID or container name> [<Repository name>[:<Tag>]]` .

    ```
    docker commit containerID/containerName repository:tag
    ```

  - For testing and restore purposes, run the image and derive a new image that has a simple name. Then, test the new image.

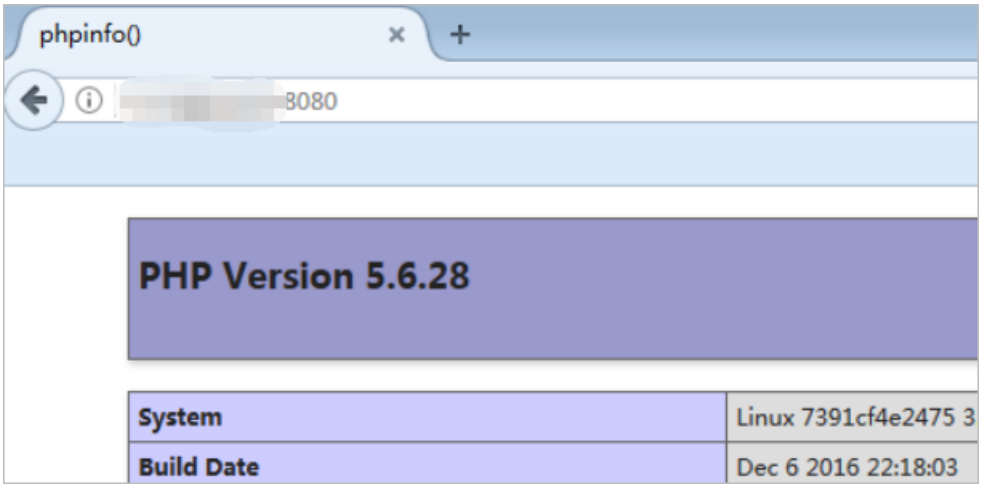
    ```
    docker commit 4c8066cd8**** apachephp:v1
    ```

- Run the container and map port 8080 of the host to the container.

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

In a browser, enter <IP address of the ECS instance>:8080 to connect to the container. A page similar to the one shown in the following figure indicates that the container runs normally.

> **Note** An inbound rule must be added to a security group of the ECS instance to allow inbound traffic on port 8080. For more information, see Add a security group rule.



Create a Docker image

1. Prepare Dockerfile.
   i. Create and edit *Dockerfile*.

```
vim Dockerfile
```

   ii. Press the /key to enter the edit mode. Add the following content to the file:

```
#Declare a base image.
FROM apachephp:v1
#Declare the owner of the base image.
MAINTAINER DTSTACK
#Specify the commands that you want to run before the container starts. You must ap
pend these commands to the end of the RUN command. Dockerfile can contain up to 127
lines. If the total length of your commands exceeds 127 lines, we recommend that yo
u write these commands to a script.
RUN mkdir /dtstact
#Specify the commands that are run on system startup. The last command must be a fr
ontend command that runs constantly. Otherwise, the container exits after all comma
nds are run.
ENTRYPOINT ping www.aliyun.com
```

   iii. Press the *Esc* key. Enter `:wq` and press the *Enter* key to save and exit *Dockerfile*.
2. Create an image.

```
docker build -t webcentos8:v1 .    # . at the end of the command line specifies the pat
h of Dockerfile and must be provided.
docker images                                  #Check whether the image is created.
```

3. Run the container and check its state.

```
docker run -d webcentos8:v1           #Run the container in the background.
docker ps                             #Query the containers that are in the running sta
te.
docker ps -a                          #Query all containers, including those in the sto
pped state.
docker logs CONTAINER ID/IMAGE        #If the container does not appear in the query re
sults, check the startup log to troubleshoot the issue based on the container ID or nam
e.
```

4. Create an image.

```
docker commit fb2844b6**** dtstackweb:v1     #Append the container ID and the name and
version number of the new image to the end of the commit parameter.
docker images                                #Query images that are downloaded and crea
ted on premises.
```

5. Push the image to a remote repository.

   By default, the image is pushed to Docker Hub. You must log on to Docker, add a tag to the image, and then name the image in the `<Docker username>/<Image name>:<Tag>` format. Then, the image is pushed to the remote repository.

```
docker login --username=dtstack_plus registry.cn-shanghai.aliyuncs.com #Enter the passw
ord of the image repository after you run this command.
docker tag [ImageId] registry.cn-shanghai.aliyuncs.com/dtstack123/test:[Tag]
docker push registry.cn-shanghai.aliyuncs.com/dtstack123/test:[Tag]
```

5.6. Deploy databases based on ECS

5.6.1. Database overview

A database is a set of data that is stored in an organized manner and can be shared by a number of users. A database provides minimal redundancy and is independent of applications. A database can be regarded as an electronic filing cabinet on which you can perform data operations such as add, query, update, and delete.

Common databases

Three types of databases are typically used:

- Oracle
  - Oracle provides a high degree of hardware stability. It can run on a variety of hardware and operating system platforms, from desktop computers to mainframes and supercomputers. Oracle supports symmetric multiprocessors, cluster multiprocessors, and large-scale processors, and works with multiple languages.

- Oracle is a multi-user system that can automatically recover from system failures in batch processing or online environments. Developer/2000 is a software development tool developed by Oracle that consists of an interactive application generator, report printer, word processor, and a centralized data dictionary. You can use these components to generate your own applications.
- Oracle presents data in two-dimensional tables and provides Structured Query Language (SQL) to implement basic database management features such as data query, modification, definition, and control.
- Data in Oracle databases can be smoothly migrated. The communication feature provided by Oracle allows programs on microcomputers to receive data from or transfer data to Oracle databases on minicomputers and mainframes.
- Oracle is a large-scale database system. It is suitable for small, medium-sized, and large application systems. It can serve both the client and server sides of server systems.

- SQL Server

  SQL Server is a relational database system provided by Microsoft. It is a scalable, high-performance database management system suitable for distributed client and server computing. SQL Server works with Windows New Technology (Windows NT) to provide a transaction-based enterprise-level information management solution. Versions earlier than SQL Server 2016 can run only on Windows.

- MySQL

  MySQL is an open source relational database management system (RDBMS) and uses the most common database management language SQL. MySQL databases can be used across platforms such as Linux and Windows.

Deployment methods

You can use ApsaraDB RDS to implement rapid deployment and lightweight O&M of databases. If you do not have an appropriate image or if you want to customize your deployment, we recommend that you manually deploy a database.

- Create and connect to an ApsaraDB RDS instance
- Manually deploy a MySQL database on an ECS Windows instance
- Manually deploy a MySQL database on an ECS instance that runs Alibaba Cloud Linux 2
- Manually deploy a MySQL database on an ECS instance that runs CentOS 8
- Manually deploy a MySQL database on an ECS instance that runs CentOS 7

5.6.2. Create and connect to an ApsaraDB RDS instance

ApsaraDB RDS is a stable, reliable, and scalable online database service. When you use an Elastic Compute Service (ECS) instance to build a business on the cloud, you can use an RDS instance to store business data. This topic describes how to create an ApsaraDB RDS for MySQL instance and connect a Linux ECS instance to the ApsaraDB RDS for MySQL instance.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

In the example, an ECS instance that has the following configurations is used. You can configure the ECS instance based on your business requirements.

- Region and zone: Hangzhou Zone I
- Instance type: ecs.g6.large
- Disk category: enhanced SSD (ESSD)
- Image: Alibaba Cloud Linux 3 64-bit public image
- Network type: Virtual Private Cloud (VPC). A public IP address is assigned to the instance.

Context

RDS is built on top of the Apsara Distributed File System and high-performance SSDs of Alibaba Cloud. RDS supports the MySQL, SQL Server, PostgreSQL, and MariaDB TX database engines. It provides a portfolio of solutions for scenarios such as disaster recovery, backup, restoration, monitoring, and migration to reduce your O&M burdens. For more information, see What is ApsaraDB RDS?

Perform the following steps to create and connect to an RDS instance:

- Step 1: Create an RDS instance
- Step 2: Create a database and a standard account for the database
- Step 3: Configure a whitelist for the RDS instance and obtain the internal endpoint and internal port number of the RDS instance
- Step 4: Connect to the ApsaraDB RDS for MySQL database from the ECS instance

Step 1: Create an RDS instance

1. Go to the ApsaraDB RDS buy page.
2. Complete the configurations of the RDS instance.

   In this example, an ApsaraDB RDS for MySQL instance is created. For more information about how to create an ApsaraDB RDS for MySQL instance, see Create an ApsaraDB RDS for MySQL instance. In this example, the following configurations are used. You can configure the RDS instance based on your business requirements.

   i. In the **Basic Configurations** step, configure the following parameters:

   - **Region**: Select China (Hangzhou).

     > **Note** If you want to transmit data between the ECS instance and the RDS instance over the internal network, you must deploy the ECS instance and the RDS instance in the same region and the same VPC. Transmission over the internal network is more secure and stable and provides higher performance than transmission over the Internet.

   - **Database Engine**: Select MySQL 8.0.
   - **Edition**: Select High-availability.
   - **Storage Type**: Select Local SSD.
   - **Zone of Primary Node**: Select Hangzhou Zone I.
   - **Deployment Method**: Select Multi-zone Deployment.
   - **Zone of Secondary Node**: Select Automatically Allocated.
   - **Instance Type**: Select rds.mysql.t1.small.

   Use the default values for other parameters.

   ii. Click **Next: Instance Configuration**.

iii. In the **Instance Configuration** step, configure the following parameters:

- **Network Type**: Select VPC.
- **VPC** and **VSwitch of Primary Node**: Select the same VPC and vSwitch as the ECS instance to which you want to connect.

Use the default values for other parameters.

iv. Click **Next: Confirm Order**. Confirm the configurations and pay for the order.

Up to 10 minutes are required to create an RDS instance. You can view and refresh the state of the RDS instance on the **Instances** page. If the state of the instance changes to **Running**, the RDS instance is created and runs normally.

Step 2: Create a database and a standard account for the database

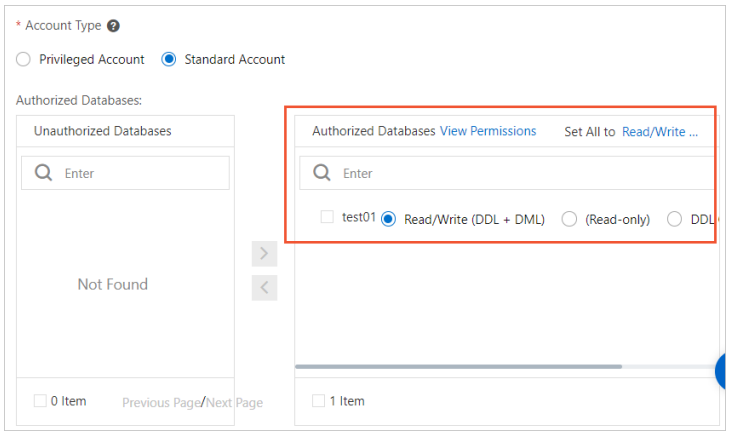
For information about how to create a database and an account for an ApsaraDB RDS for MySQL instance, see Create databases and accounts for an ApsaraDB RDS for MySQL instance. In this example, a database named `test01` is created for the RDS instance and a standard account named `testuser01` is created to log on to the `test01` database.

1. Go to the Instances page.
2. In the top navigation bar, select the **China (Hangzhou)** region.
3. Find and click the ID of the RDS instance that you created.
4. Create a database on the RDS instance.
   i. In the left-side navigation pane of the Basic Information page, click **Databases** and then click **Create Database**.
   ii. In the **Create Database** dialog box, configure the following parameters and click **Create**:
   - **Database Name**: Specify a name for the database. Example: `test01`.
   - **Supported Character Set**: Select utf8.

   Use the default values for other parameters.
5. Create a standard account.
   i. In the left-side navigation pane of the Basic Information page, click **Accounts**. On the **Accounts** tab, click **Create Account**.

ii. In the **Create Account** panel, configure the following parameters and click **OK**:

- **Database Account**: Specify a name for the account. Example: `testuser01`.
- **Account Type**: Select Standard Account.
- Authorized Databases: After you select **Standard Account**, the **Authorized Databases** section appears. You must grant the `testuser01` account read and write permissions on the `test01` database.



- Password and Confirm Password: Specify a password. We recommend that you set a complex password to improve data security. Keep your password confidential.

Use the default values for other parameters.

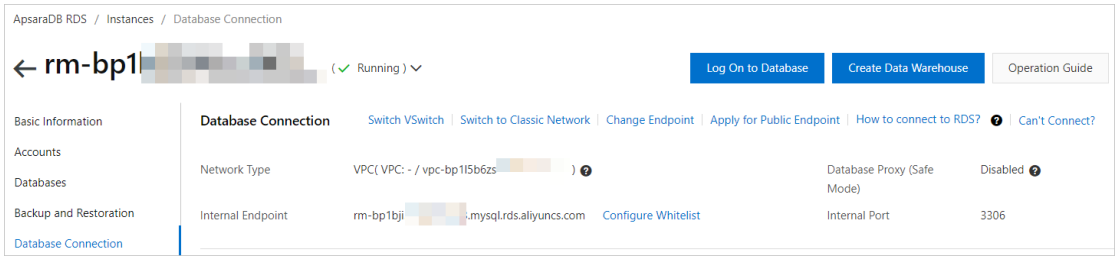
Step 3: Configure a whitelist for the RDS instance and obtain the internal endpoint and internal port number of the RDS instance

Before you obtain the internal endpoint and internal port number of the RDS instance, you must check whether the ECS instance and the RDS instance meet the conditions for communication over the internal network. For more information, see the "Step 1: Check whether your application can connect to the RDS instance over an internal network" section in Use a database client or the CLI to connect to an ApsaraDB RDS for MySQL instance. In this example, the ECS instance and the RDS instance reside within the same VPC and are able to communicate with each other over the internal network.

1. Go to the Instances page.
2. In the top navigation bar, select the **China (Hangzhou)** region.
3. Find and click the ID of the RDS instance that you created.
4. Configure a whitelist for the RDS instance.
   i. In the left-side navigation pane of the Basic Information page, click **Data Security**.
   ii. On the **Whitelist Settings** tab, find the whitelist named **default** and click **Modify** on the right side.
   iii. In the **Edit Whitelist** dialog box, remove the default `127.0.0.1` from the **IP Addresses** field and add *<Internal IP address of the ECS instance>*.

   For more information about how to view the IP addresses of an ECS instance, see View information of instances on the Instances page.
   iv. Click **OK**.
5. Obtain the internal endpoint and internal port number of the RDS instance.
   i. In the left-side navigation pane of the Basic Information page, click **Database Connection**.

ii. On the **Database Connection** page, view the **Internal Endpoint** and **Internal Port** values.



You must save the internal endpoint and internal port number for the subsequent connections to the database.

Step 4: Connect to the ApsaraDB RDS for MySQL database from the ECS instance

1. Log on to the Linux ECS instance.

   For more information, see Connection methodsGuidelines on instance connection.

2. (Optional) Install MySQL on the ECS instance.

   If MySQL is not installed on your ECS instance, install MySQL first. For example, you can run the following command to install MySQL on an ECS instance that runs Alibaba Cloud Linux or CentOS:

   ```
   yum -y install mysql
   ```

3. Run the following command to connect to the MySQL database:

   ```
   mysql -h <Internal endpoint of the RDS instance> -P <Internal port number of the RDS in
   stance> -u<Standard account of the RDS instance> -p
   ```

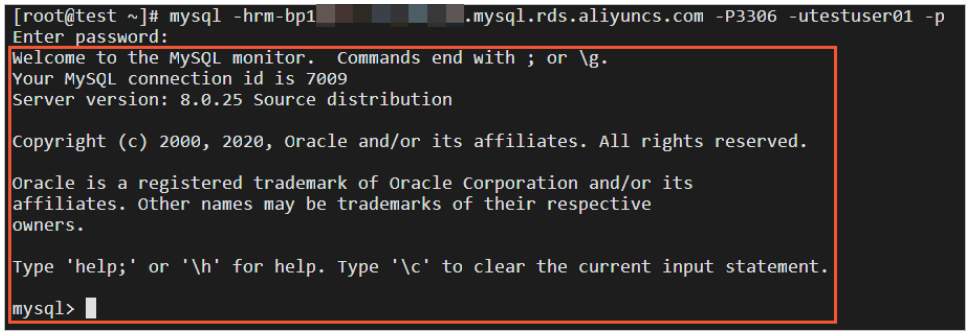
   Variables:

   - *<Internal endpoint of the RDS instance>*: the internal endpoint of the RDS instance. Example: `rm-bp1bj****.mysql.rds.aliyuncs.com`.
   - *<Internal port number of the RDS instance>*: the internal port number of the RDS instance. For example, the default port number of the MySQL database is 3306.
   - *<Standard account of the RDS instance>*: the standard account of the RDS instance. In this example, `testuser01` is used.

4. In the `Enter password:` command line, enter the password of `testuser01`.

   > **Note**

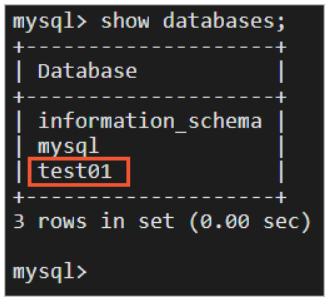
   The following figure shows that you are connected to the MySQL database. If the connection fails, see the "Common connection errors" section in Common connection errors.



5. (Optional) Run the following command in the MySQL client to view the details of the database:

```
show databases;
```

You can view the `test01` database in the command output.



References

Getting Started

5.6.3. Manually deploy a MySQL database on an ECS instance that runs Alibaba Cloud Linux 2

Build a database in Alibaba CloudDeploy MySQL on a virtual machine

MySQL is a relational database management system and is used to build the LAMP or LNMP environment. This topic describes how to install, configure, and access a MySQL database on an Elastic Compute Service (ECS) instance that runs Alibaba Cloud Linux 2.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

In this topic, an ECS instance that has the following configurations is used:

- Instance type: ecs.c6.large
- Image: Alibaba Cloud Linux 2.1903 LTS 64-bit public image
- Network type: Virtual Private Cloud (VPC). A public IP address is assigned to the instance.

Context

In the example, MySQL 8.0.28 is deployed. The version of MySQL may vary based on the update of software repositories. The following MySQL installation paths are used:

- Configuration file: */etc/my.cnf*
- Data storage: */var/lib/mysql*

- Command files: */usr/bin* and */usr/sbin*

Step 1: Install MySQL

1. Connect to the ECS instance.

   For more information, see Connection methodsGuidelines on instance connection.

2. Run the following command to update the YUM repository:

   ```
   sudo rpm -Uvh https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm
   ```

3. Run the following command to install MySQL:

   ```
   sudo yum -y install mysql-community-server --enablerepo=mysql80-community --nogpgcheck
   ```

4. Run the following command to check the version of MySQL:

   ```
   mysql -V
   ```

   The following example command output indicates that MySQL is installed.

   ```
   [root@test ~]# mysql -V
   mysql  Ver 8.0.28 for Linux on x86_64 (MySQL Community Server - GPL)
   [root@test ~]#
   ```

Step 2: Configure MySQL

1. Run the following command to start MySQL:

   ```
   systemctl start mysqld
   ```

2. Run the following command to enable MySQL to run on system startup:

   ```
   systemctl enable mysqld
   ```

3. Run the following command to check the */var/log/mysqld.log* file and obtain and record the initial password of the root user:

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

   A command output similar to the following one is returned:

   ```
   2022-02-14T09:27:18.470008Z 6 [Note] [MY-010454] [Server] A temporary password is gener
   ated for root@localhost: r_V&f2wyu_vI
   ```

   Note `r_V&f2wyu_vI` at the end of the command output is the initial password, which is required to configure the security settings for MySQL.

4. Run the following command to configure the security settings for MySQL:

   ```
   mysql_secure_installation
   ```

i. Reset the password of the root user.

> Note

```
Enter password for user root: # Enter the initial password of the root user that yo
u obtained.
The existing password for the user account root has expired. Please set a new passw
ord.
New password: # Enter a new password.
Re-enter new password: # Enter the new password again.
The 'validate_password' component is installed on the server.
The subsequent steps will run with the existing configuration
of the component.
Using existing password for root.
Change the password for root ? (Press y|Y for Yes, any other key for No) : Y # Ente
r Y to update the password. You can also enter N to skip updating the password.
New password: # Enter the new password.
Re-enter new password: # Enter the new password again.
Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any other ke
y for No) :Y # Enter Y to use the new password.
```

ii. Delete the anonymous user.

```
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 # Enter Y to
delete the default anonymous user.
Success.
```

iii. Deny remote access by the root user.

```
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 # Enter
Y to deny remote access by the root user.
Success.
```

iv. Delete the test database and the access permissions on the database.

```
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 # Enter Y to delete the test database and the access permissions on the database.

 - Dropping test database...
Success.
 - Removing privileges on test database...
Success.
```

v. Reload privilege tables.

```
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 # Enter
Y to reload privilege tables.
Success.
All done!
```

For more information about the security settings of MySQL, see MySQL documentation.

Step 3. Access the MySQL database

You can use a database client or Data Management Service (DMS) provided by Alibaba Cloud to access the MySQL database. In this topic, DMS is used to access the MySQL database.

1. Add an inbound rule to a security group of the ECS instance to allow traffic on the port number of the MySQL database.

   The default port number of the MySQL database is 3306. You must add inbound rules to a security group of the ECS instance and allow traffic on port 3306. For more information, see Add a security group rule.

2. On the ECS instance, create an account for remote logon to the MySQL database.

   i. Run the following command and enter the password for the root user to log on to the MySQL database:

   ```
   mysql -uroot -p
   ```

ii. Run the following commands in sequence to create an account for remote logon to the MySQL database.

We recommend that you use a non-root account to log on to the MySQL database. In this example, the `dms` account and the `123456` password are used.

> **Notice** When you create an account, replace the `123456` password with a valid password and keep it confidential. The password must be 8 to 30 characters in length and must contain uppercase letters, lowercase letters, digits, and special characters. Special characters include
>
> `( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /`

```
mysql> create user 'dmsTest'@'%' identified by '123456'; # Create a database accoun
t named dmsTest and grant the remote connection permissions to the account.
mysql> grant all privileges on *.* to 'dmsTest'@'%'; # Grant the dmsTest account al
l permissions on the database.
mysql> flush privileges; # Refresh the permissions.
```

3. Log on to the DMS console.
4. In the top navigation bar of the **Data Assets** tab, click **Instances**.
5. On the **Instances** tab, click **+ New**.
6. In the **Add Instance** dialog box, configure the parameters.
   i. Click **ECS Self-built** and then click **MySQL**.
   ii. In the **Basic Information** section, configure the following parameters to access the database:
   - **Data Source**: Select ECS Self-built.
   - **Database Type** :Select MySQL.
   - **Instance Region** and **ECS Instance ID**: Set the parameters based on the information of the ECS instance. For more information about how to obtain the information of an ECS instance, see View information of instances on the Instances page.
   - **Port**: Use the default value 3306.
   - **Database Account**: Enter the database account for remote connection. In this example, enter `dmsTest`.
   - **Database Password**: Enter the password of the database account for remote connection.

   Use the default values for other parameters.
   iii. Click **Test Connection**.

   If you are connected to the MySQL database, the **Success** dialog box appears.
   iv. Click **Submit**.
7. After you submit the configurations, go to the Database List tab on the Home page as prompted. You can use Shortcuts in the DNS console to manage the database. For more information about the operations on DMS, see Overview.

5.6.4. Manually deploy a MySQL database on an ECS instance that runs CentOS 8

MySQL is a relational database management system and is often used to build the LAMP or LNMP environment. This topic describes how to install, configure, and connect to a MySQL database on an Elastic Compute Service (ECS) instance that runs CentOS 8.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

In this topic, an ECS instance that has the following configurations is used:

- Instance type: ecs.c6.large
- Operating system: CentOS 8.2 64-bit public image
- Network type: Virtual Private Cloud (VPC). A public IP address is assigned to the instance.

Context

In the example, MySQL 8.0.21 is deployed. The version of MySQL may vary based on the update of software repositories. The following MySQL installation paths are used:

- Configuration file: */etc/my.cnf*
- Data storage: */var/lib/mysql*
- Command files: */usr/bin* and */usr/sbin*

Step 1: Install MySQL

1. Connect to the ECS instance that runs CentOS 8.

   For more information,see Connect to a Linux instance by using a password.
2. Change the CentOS 8 repository address.
3.
4. Run the following command to view the MySQL version:

   ```
   mysql -V
   ```

   A command output similar to the following one indicates the MySQL version.

   ```
   [root@test ~]# mysql -V
   mysql  Ver 8.0.21 for Linux on x86_64 (Source distribution)
   ```

Step 2: Configure MySQL

1.
2.
3. Run the following command to configure security settings for MySQL and set the password:

   ```
   mysql_secure_installation
   ```

   After you run the command, perform the following operations based on the command prompts:

   i. Enter *Y* and press the Enter key to make the configurations.

ii. Enter *2* as the password strength and press the Enter key.

0 indicates a low password strength, *1* indicates a medium password strength, and *2* indicates a high password strength. We recommend that you use a high password strength.

iii. Enter a new password for MySQL and confirm it.

In this example, the password is set to `PASSword123!`.

iv. Enter *Y* and press the Enter key to use the password.

v. Enter *Y* and press the Enter key to delete anonymous users.

vi. Enter *Y* and press the Enter key to disallow the root account to connect to MySQL.

vii. Enter *Y* and press the Enter key to delete the `test` database and the access permissions on the `test` database.

viii. Enter *Y* and press the Enter key to reload privilege tables.

Step 3: Connect to the MySQL database

We recommend that you use a non-root account to connect to the MySQL database. In this example, an account is created to connect to MySQL.

1. Add an inbound rule to a security group of the ECS instance to allow traffic on the port required by MySQL.

   The default port number of the MySQL database is 3306. You must add inbound rules to a security group of the ECS instance and allow traffic on port 3306. For more information, see Add a security group rule.

2. Create and configure an account to connect to MySQL on the instance.

   i. Run the following command and enter the password for the root user to log on to the MySQL database:

   ```
   mysql -uroot -p
   ```

   ii. Run the following commands in sequence to create an account to connect to MySQL on the MySQL client and allow the instance to connect to MySQL by using this account.

   In this example, the account is named `dms` and the password is set to `PASSword123!`.

   ```
   create user 'dms'@'%' identified by 'PASSword123!';
   grant all privileges on *.* to 'dms'@'%'with grant option;
   flush privileges;
   ```

   > **Note** When you create an account, the password must meet the following requirements: The password must be 8 to 30 characters in length and must contain uppercase letters, lowercase letters, digits, and special characters. Special characters include
   >
   > `( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /`

3. Connect to MySQL by using the `dms` account.

   - We recommend that you use Data Management Service (DMS) provided by Alibaba Cloud to connect to the MySQL database. For more information, see Register an ApsaraDB instance.
   - You can use connection tools such as MySQL Workbench and Navicat on your computer to

connect to the MySQL database.

5.6.5. Manually deploy a MySQL database on an ECS instance that runs CentOS 7

Build a database in Alibaba CloudDeploy MySQL on a virtual machine

MySQL is a relational database management system and is used to build the LAMP or LNMP environment. This topic describes how to install, configure, and access a MySQL database on a Linux Elastic Compute Service (ECS) instance.

Prerequisites

An ECS instance is created. For more information, see Create an instance by using the wizard.

In this topic, an ECS instance that has the following configurations is used:

- Instance type: ecs.c6.large
- Image: CentOS 7.8 64-bit public image
- Network type: Virtual Private Cloud (VPC). A public IP address is assigned to the instance.

Context

In the example, MySQL 8.0.28 is deployed. The version of MySQL may vary based on the update of software repositories. The following MySQL installation paths are used:

- Configuration file: */etc/my.cnf*
- Data storage: */var/lib/mysql*
- Command files: */usr/bin* and */usr/sbin*

Step 1: Install MySQL

1. Connect to the ECS instance.

   For more information, see Connection methodsGuidelines on instance connection.

2. Run the following command to update the YUM repository:

   ```
   sudo rpm -Uvh https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm
   ```

3. Run the following command to install MySQL:

   ```
   sudo yum -y install mysql-community-server --enablerepo=mysql80-community --nogpgcheck
   ```

4. Run the following command to check the version of MySQL:

   ```
   mysql -V
   ```

   The following example command output indicates that MySQL is installed.

   ```
   [root@test ~]# mysql -V
   mysql  Ver 8.0.28 for Linux on x86_64 (MySQL Community Server - GPL)
   [root@test ~]#
   ```

Step 2: Configure MySQL

1. Run the following command to start MySQL:

```
systemctl start mysqld
```

2. Run the following command to enable MySQL to run on system startup:

```
systemctl enable mysqld
```

3. Run the following command to check the */var/log/mysqld.log* file and obtain and record the initial password of the root user:

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

A command output similar to the following one is returned:

```
2022-02-14T09:27:18.470008Z 6 [Note] [MY-010454] [Server] A temporary password is gener
ated for root@localhost: r_V&f2wyu_vI
```

> **Note** `r_V&f2wyu_vI` at the end of the command output is the initial password, which is required to configure the security settings for MySQL.

4. Run the following command to configure the security settings for MySQL:

```
mysql_secure_installation
```

i. Reset the password of the root user.

> **Note**

```
Enter password for user root: # Enter the initial password of the root user that yo
u obtained.
The existing password for the user account root has expired. Please set a new passw
ord.
New password: # Enter a new password.
Re-enter new password: # Enter the new password again.
The 'validate_password' component is installed on the server.
The subsequent steps will run with the existing configuration
of the component.
Using existing password for root.
Change the password for root ? (Press y|Y for Yes, any other key for No) : Y # Ente
r Y to update the password. You can also enter N to skip updating the password.
New password: # Enter the new password.
Re-enter new password: # Enter the new password again.
Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any other ke
y for No) :Y # Enter Y to use the new password.
```

ii. Delete the anonymous user.

```
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 # Enter Y to
delete the default anonymous user.
Success.
```

iii. Deny remote access by the root user.

```
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 # Enter
Y to deny remote access by the root user.
Success.
```

iv. Delete the test database and the access permissions on the database.

```
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 # Enter Y to delete the test database and the access permissions on the database.

 - Dropping test database...
Success.
 - Removing privileges on test database...
Success.
```

v. Reload privilege tables.

```
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 # Enter
Y to reload privilege tables.
Success.
All done!
```

For more information about the security settings of MySQL, see MySQL documentation.

Step 3. Access the MySQL database

You can use a database client or Data Management Service (DMS) provided by Alibaba Cloud to access the MySQL database. In this topic, DMS is used to access the MySQL database.

1. Add an inbound rule to a security group of the ECS instance to allow traffic on the port number of the MySQL database.

   The default port number of the MySQL database is 3306. You must add inbound rules to a security group of the ECS instance and allow traffic on port 3306. For more information, see Add a security group rule.

2. On the ECS instance, create an account for remote logon to the MySQL database.
   i. Run the following command and enter the password for the root user to log on to the MySQL database:

   ```
   mysql -uroot -p
   ```

   ii. Run the following commands in sequence to create an account for remote logon to the MySQL database.

   We recommend that you use a non-root account to log on to the MySQL database. In this example, the `dms` account and the `123456` password are used.

   > **Notice** When you create an account, replace the `123456` password with a valid password and keep it confidential. The password must be 8 to 30 characters in length and must contain uppercase letters, lowercase letters, digits, and special characters. Special characters include
   > `( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ' < > , . ? /`

   ```
   mysql> create user 'dmsTest'@'%' identified by '123456'; # Create a database accoun
   t named dmsTest and grant the remote connection permissions to the account.
   mysql> grant all privileges on *.* to 'dmsTest'@'%'; # Grant the dmsTest account al
   l permissions on the database.
   mysql> flush privileges; # Refresh the permissions.
   ```

3. Log on to the DMS console.
4. In the top navigation bar of the **Data Assets** tab, click **Instances**.
5. On the **Instances** tab, click **+ New**.
6. In the **Add Instance** dialog box, configure the parameters.
   i. Click **ECS Self-built** and then click **MySQL**.
   ii. In the **Basic Information** section, configure the following parameters to access the database:
   - **Data Source**: Select ECS Self-built.
   - **Database Type** :Select MySQL.
   - **Instance Region** and **ECS Instance ID**: Set the parameters based on the information of the ECS instance. For more information about how to obtain the information of an ECS instance, see View information of instances on the Instances page.
   - **Port**: Use the default value 3306.
   - **Database Account**: Enter the database account for remote connection. In this example, enter `dmsTest`.
   - **Database Password**: Enter the password of the database account for remote connection.

   Use the default values for other parameters.
   iii. Click **Test Connection**.

   If you are connected to the MySQL database, the **Success** dialog box appears.
   iv. Click **Submit**.
7. After you submit the configurations, go to the Database List tab on the Home page as prompted.

You can use Shortcuts in the DNS console to manage the database. For more information about the operations on DMS, see Overview.

5.6.6. Manually deploy a MySQL database on an ECS Windows instance

This topic describes how to manually deploy a MySQL database on an Elastic Compute Service (ECS) Windows instance.

Prerequisites

Procedure

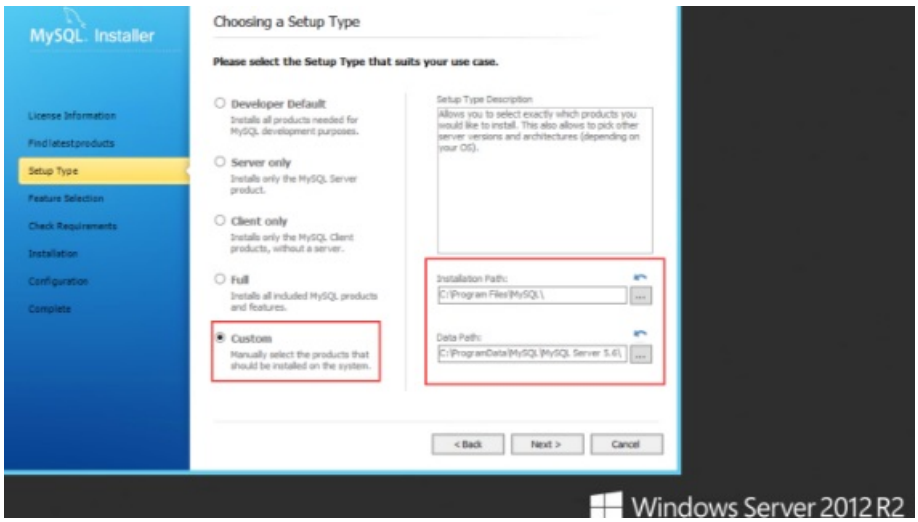
1. Create an ECS instance that runs Windows Server 2012. For more information, see Create an instance by using the wizard.
2.
3. Download and install the *vcredist_x86.exe* plug-in.
4. Download an MySQL installation package from the MySQL official website.

   In this topic, the MySQL 5.6.15.0 installation package is used.
5. Install MySQL on the ECS instance.
   i. Double-click *mysql-installer-community-5.6.15.0.msi* to install MySQL.

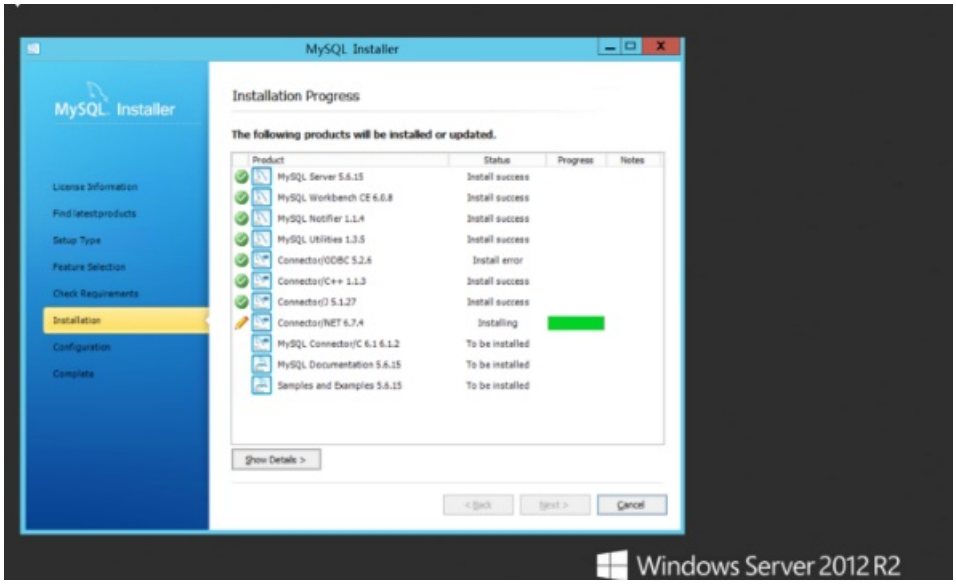
ii. Select **Install MySQL Products**.

iii. Select I accept the license terms and skip the update check, click Next, and then select **Custom**. In the right pane, specify the MySQL installation location and the database location, and click **Next**.

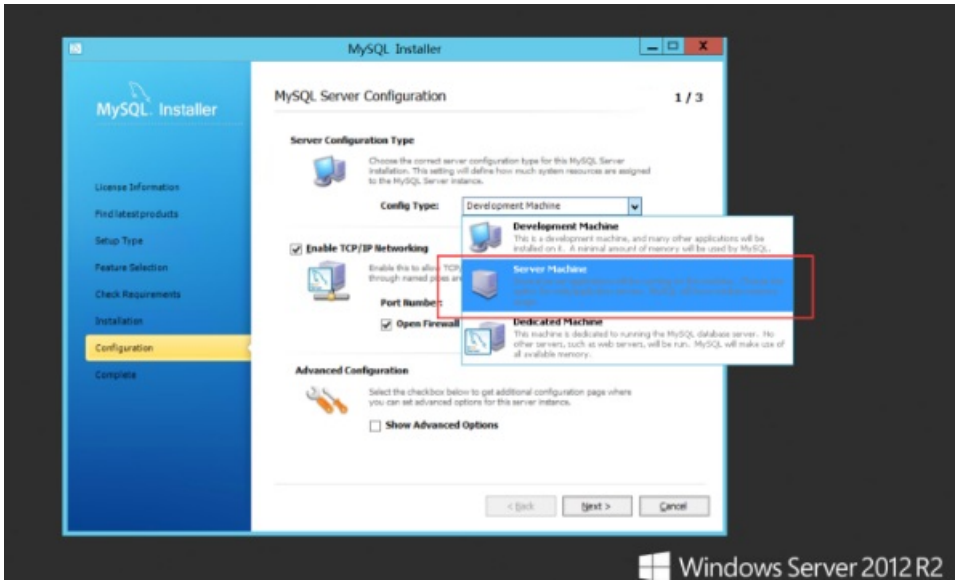
In this example, the default MySQL installation location and database location are used.



iv. Keep the default values unchanged, click **Next**, and then click **Execute** to start to install MySQL.



v. Click **Next** to go to the MySQL Server Configuration page, and select **Server Machine** from the Config Type drop-down list.



vi. Keep the default values unchanged, click **Next**, and then enter the password of the root user to complete the installation of MySQL.

After you install MySQL, the MySQL Command-Line Client icon appears on the start page.

6. Add inbound rules to the security group of the ECS instance and allow traffic on port 3306. For more information, see Add a security group rule.

5.6.7. Manage self-managed databases on ECS instances

Self-managed databases on Elastic Compute Service (ECS) instances are databases that are installed and configured on ECS instances. You can use Data Management Service (DMS) to manage self-managed databases on ECS instances. This topic describes how to use DMS to add and manage self-managed databases on ECS instances. In the example, a MySQL database is used.

Prerequisites

1. DMS is activated. For more information, see Activate DMS.
2. An ECS instance of a memory-optimized r-series instance family is created. For more information about how to create an ECS instance, see Create an instance by using the wizard.
3. An inbound rule is added to the security group of the ECS instance to allow traffic for the MySQL database on the listening port. The default listening port number for the MySQL database is 3306.
4. A MySQL database is installed on the ECS instance. For more information, see Database overview.
5. A non-root account is created for the MySQL database on the ECS instance.

   > **Note** By default, MySQL databases do not allow remote access from the root account. If you have changed the default settings to allow remote access from the root account, skip this step.

   In this example, run the following command to create an account named `dms` for logging on to the MySQL database, set the password to `Ecs123!`, and grant all permissions to the account:

   ```
   create user 'dms'@'%' IDENTIFIED BY 'Ecs123!';
   grant all privileges on *.* to 'dms'@'%' with grant option;
   flush privileges;
   ```

Context

DMS is a database management web terminal provided by Alibaba Cloud to help manage self-managed databases on ECS instances that run Windows or Linux. You can add self-managed databases in the DMS console to use data management features such as creating databases and tables. DMS supports self-managed MySQL, SQL Server, PostgreSQL, MongoDB, and Redis databases. For more information, see DMS documentation.

Procedure

1. Log on to the DMS console.
2. In the top navigation bar, click **Instances**.
3. On the **Instance List** tab, click **New**.
4. In the **Add Instance** dialog box, click the ECS Self-built tab, click MySQL, and then configure parameters for the self-managed database.

   The following table describes the parameters.

Parameter	Description
Data Source	Select ECS Self-built.
Database Type	The type of the self-managed database on the ECS instance. In this example, select *MySQL*.
Instance Region	The region where the ECS instance is located.
ECSInstance ID	The ID of the ECS instance.
Port	The number of the listening port used to access the database on the ECS instance. In this example, set Port to *3306*.
Database Account	The account used to log on to the self-managed database on the ECS instance. In this example, the account used to log on to the MySQL database is `dms`. **Note** We recommend that you use a non-root account.
Database password	The password used to log on to the self-managed database on the ECS instance. In this example, the password of *dms* is `Ecs123!`.
Control Mode	In this example, select **Flexible Management**. For more information about the control mode, see Control modes.

5. Click **Submit**.
6. Log on to the database.

What's next

After you log on to the database, you can perform operations on the database by using SQL statements or the DMS console.

Related information

- Migrate a local user-created database to ECS
- Migrate data between user-created databases on ECS instances

5.7. Build a primary/secondary PostgreSQL architecture

PostgreSQL is one of the most advanced open source databases and supports NoSQL data types such as JSON, XML, and hstore. This topic describes how to build a primary/secondary PostgreSQL architecture on an Elastic Compute Service (ECS) instance that runs CentOS 7.

Prerequisites

-
- An inbound rule is added to the security group of the instance to allow traffic on port 5432. For more information, see Add a security group rule.

Context

The procedure described in this topic is applicable to Alibaba Cloud users who are familiar with Alibaba Cloud ECS, Linux operating systems, and PostgreSQL databases.

In this topic, the following instance type and software versions are used. The operations may vary based on your instance type and software versions.

- Instance type: ecs.g6.large
- Operating system: CentOS 7.2
- PostgreSQL: 9.6

To use YUM to install PostgreSQL and build a primary/secondary PostgreSQL architecture, perform the following operations:

Step 1: Create two ECS instances

To build a primary/secondary PostgreSQL architecture, you must create two instances of the Virtual Private Cloud (VPC) type. One instance works as the primary node, and the other instance works as the secondary node. For more information, see Create an instance by using the wizard.

> **Note** We recommend that you do not assign public IP addresses to the ECS instances. You can bind an elastic IP address (EIP) to each ECS instance. This allows you to upgrade the configurations or optimize the architecture based on your requirements. For more information, see Apply for an EIP.

Step 2: Configure the primary node of PostgreSQL

1. Run the following commands in sequence to install PostgreSQL:

```
yum update -y
```

```
wget --no-check-certificate https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

```
rpm -ivh pgdg-redhat-repo-latest.noarch.rpm
```

```
yum install postgresql96-server postgresql96-contrib -y
```

```
/usr/pgsql-9.6/bin/postgresql96-setup initdb
```

> **Note** In this topic, *PostgreSQL 9.6* is used. We recommend that you use the latest version of PostgreSQL.

2. Run the following commands in sequence to start PostgreSQL and enable PostgreSQL to run on system startup:

```
systemctl start postgresql-9.6.service    #Start PostgreSQL.
```

```
systemctl enable postgresql-9.6.service   #Enable PostgreSQL to run on system startup.
```

3. Create a database account named replica. This database is used for replication between the primary and secondary nodes. Then, specify the password, logon permission, and backup permission.
   i. Run the following command to log on to PostgreSQL by using the postgres account:

   ```
   su - postgres
   ```

   ii. When `-bash-4.2$` appears, you are logged on to PostgreSQL. Then, run the following command to go to the interactive terminal of PostgreSQL:

   ```
   psql
   ```

   iii. When `postgres=#` appears, you are accessing the PostgreSQL interactive terminal. Execute the following statement to set a password for the `postgres` account to enhance security:

   ```
   ALTER USER postgres WITH PASSWORD 'YourPassWord';
   ```

   iv. Execute the following SQL statement to create a database account named replica, specify a password, and configure the logon and backup permissions for the account.

   In this example, the password is set to `replica`.

   ```
   CREATE ROLE replica login replication encrypted password 'replica';
   ```

   v. Execute the following statement to check whether the database account is created:

   ```
   SELECT usename from pg_user;
   ```

   The following results indicate that the account named replica is created:

   ```
   usename
   ----------
   postgres
   replica
   (2 rows)
   ```

vi. Execute the following statement to check whether the permissions are configured:

```
SELECT rolname from pg_roles;
```

The following results indicate that the permissions are configured:

```
rolname
----------
postgres
replica
(2 rows)
```

vii. Run the following command and press the `Enter` key to exit the PostgreSQL interactive terminal:

```
\q
```

viii. Run the following command and press the `Enter` key to exit PostgreSQL.

```
exit
```

4. Run the following command to open the *pg_hba.conf* file, and then configure a whitelist for replica:

```
vim /var/lib/pgsql/9.6/data/pg_hba.conf
```

Add the following lines to the `IPv4 local connections` field:

```
host    all             all             <IPv4 CIDR block of the secondary node>
md5     #Enable MD5 password encryption for connections in the CIDR block of the VPC.
host    replication     replica         <IPv4 CIDR block of the secondary node>
md5     #Enable data synchronization from the replication database.
```

After the lines are added, press the *Esc* key, enter `:wq`, and then press the *Enter* key to save the file and exit.

5. Run the following command to open the *postgresql.conf* file:

```
vim /var/lib/pgsql/9.6/data/postgresql.conf
```

Find and modify the following parameters:

```
listen_addresses = '*'   #Specify the IP addresses on which the server listens for conn
ections from client applications.
wal_level = hot_standby  #Enable the hot standby mode.
synchronous_commit = on  #Enable synchronization.
max_wal_senders = 32     #Specify the maximum number of synchronization processes.
wal_sender_timeout = 60s #Specify the timeout value for the streaming replication insta
nce to synchronize data.
max_connections = 100    #Specify the maximum number of connections. The value of max_c
onnections for the secondary node must be greater than that for the primary node.
```

After the parameters are modified, press the *Esc* key, enter `:wq`, and then press the *Enter* key to save the file and exit.

6. Run the following command to restart the PostgreSQL service:

```
systemctl restart postgresql-9.6.service
```

Step 3: Configure the secondary node of PostgreSQL

1. Run the following commands in sequence to install PostgreSQL:

```
yum update -y
```

```
wget --no-check-certificate https://download.postgresql.org/pub/repos/yum/reporpms/EL-7
-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

```
rpm -ivh pgdg-redhat-repo-latest.noarch.rpm
```

```
yum install postgresql96-server postgresql96-contrib -y
```

2. Run the following command to create a backup directory by using the pg_basebackup utility:

```
# pg_basebackup -D /var/lib/pgsql/9.6/data -h <IP address of the primary node> -p 5432
-U replica -X stream -P
```

In this example, the `Password` parameter is set to `replica`.

```
Password:
30075/30075 kB (100%), 1/1 tablespace
```

3. Run the following commands in sequence to create and modify the *recovery.conf* file:

```
cp /usr/pgsql-9.6/share/recovery.conf.sample /var/lib/pgsql/9.6/data/recovery.conf
```

```
vim /var/lib/pgsql/9.6/data/recovery.conf
```

Find and modify the following parameters:

```
standby_mode = on     #Declare the secondary node.
primary_conninfo = 'host=<IP address of the primary node> port=5432 user=replica passwo
rd=replica' #Specify the connection information of the primary node.
recovery_target_timeline = 'latest' #Synchronize the latest data by using streaming rep
lication.
```

After the parameters are modified, press the *Esc* key, enter `:wq`, and then press the *Enter* key to save the file and exit.

4. Run the following command to open the *postgresql.conf* file:

```
vim /var/lib/pgsql/9.6/data/postgresql.conf
```

Find and modify the following parameters:

```
max_connections = 1000             # Specify the maximum number of connections. The val
ue for the secondary node must be greater than that for the primary node.
hot_standby = on                   # Enable the hot standby mode.
max_standby_streaming_delay = 30s  #Specify the maximum delay for streaming replication
.
wal_receiver_status_interval = 1s  #Specify the maximum interval for the secondary node
to report its running state to the primary node.
hot_standby_feedback = on          # Enable the secondary node to report errors to the
primary node during replication.
```

After the parameters are modified, press the *Esc* key, enter `:wq`, and then press the *Enter* key to save the file and exit.

5. Run the following command to modify the group and owner of the data directory:

```
chown -R postgres.postgres /var/lib/pgsql/9.6/data
```

6. Run the following commands in sequence to start PostgreSQL and enable PostgreSQL to run on system startup:

```
systemctl start postgresql-9.6.service    #Start PostgreSQL.
```

```
systemctl enable postgresql-9.6.service   #Enable PostgreSQL to run on system startup.
```

Step 4: Test the primary/secondary PostgreSQL architecture

To test the primary/secondary PostgreSQL architecture, make sure that data can interact between the primary and secondary nodes. For example, the following content shows the expected test result when you create a backup directory for the secondary node:

```
# pg_basebackup -D /var/lib/pgsql/9.6/data -h <IP address of the primary node> -p 5432 -U r
eplica -X stream -P
```

1. Run the following command to check the sender process on the primary node:

```
ps aux |grep sender
```

The following command output indicates that the sender process is available:

```
postgres  2916  0.0  0.3 340388  3220 ?        Ss   15:38   0:00 postgres: wal sender
process replica 192.168.**.**(49640) streaming 0/F01C1A8
```

2. Run the following command to check the receiver process on the secondary node:

```
ps aux |grep receiver
```

The following command output indicates that the receiver process is available:

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

3. On the primary node, go to the PostgreSQL interactive terminal and execute the following SQL statement to check the state of the secondary node:

```
select * from pg_stat_replication;
```

The following results indicate that the state of the secondary node can be checked:

```
pid | usesysid | usename | application_name | client_addr | client_hostname | client_po
rt | backend_start | backend_xmin | state | sent_location | write_locati
on | flush_location | replay_location | sync_priority | sync_state
------+----------+---------+------------------+--------------+-----------------+------
------- +-------------------------------+-------------+-----------+---------------+---
----------
---+----------------+-----------------+--------------+------------
2916 | 16393 | replica | walreceiver | 192.168.**.** | | 49640 | 2017-05-02 15:38:06.18
8988+08 | 1836 | streaming | 0/F01C0C8 | 0/F01C0C8
| 0/F01C0C8 | 0/F01C0C8 | 0 | async
(1 rows)
```

5.8. Deploy RabbitMQ

RabbitMQ is an open source message broker that implements the Advanced Message Queuing Protocol (AMQP) to store and forward messages in a distributed system. RabbitMQ is easy to use, scalable, and highly available. This topic describes how to deploy RabbitMQ on an Elastic Compute Service (ECS) instance.

Prerequisites

A security group of the virtual private cloud (VPC) type is created. An inbound rule that allows traffic on ports 80, 5672, and 15672 is added to the security group. If you want to connect to a Linux instance in the security group by using Secure Shell (SSH), you must configure the rule to allow traffic on port 22. For more information, see Add a security group rule.

Context

The RabbitMQ server is written in the Erlang programming language. RabbitMQ supports multiple types of clients such as Python, Ruby, .NET, Java, Java Message Service (JMS), C, Professional Hypertext Preprocessor (PHP), ActionScript, Extensible Messaging and Presence Protocol (XMPP), Simple Text Oriented Messaging Protocol (STOMP), and Asynchronous JavaScript and XML (AJAX).

Manually deploy RabbitMQ. This method is suitable for users who have a basic knowledge of Linux commands and can perform personalized deployment. Use the following operating system and software versions to manually deploy RabbitMQ:

- Operating system: CentOS 7.8 64-bit public image
- RabbitMQ: RabbitMQ 3.7.8
- Erlang: Erlang 21.1
- JDK: JDK 1.8.0_282

Manually deploy RabbitMQ

1. Create and connect to a Linux instance.

i. Create a Linux instance.

For more information, see Create an instance by using the wizard. When you configure parameters to create the instance, take note of the following items:

- Select Assign Public IPv4 Address in the Public IP Address section in the Networking step.
- Select the security group configured in the "Prerequisites" section.
- Complete other settings.

ii. Connect to the instance.

For more information, see Connect to a Linux instance by using a password.

2. Install Erlang.

i. Run the following command to install the dependency for Erlang:

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

ii. Run the following command to download the Erlang installation package:

```
wget http://erlang.org/download/otp_src_21.1.tar.gz
```

iii. Run the following command to decompress the Erlang installation package:

```
tar -zxvf otp_src_21.1.tar.gz
```

iv. Run the following commands to go to the path to which the Erlang installation package is decompressed and create a directory for Erlang:

```
cd otp_src_21.1
mkdir -p /usr/local/erlang
```

v. Run the following commands in sequence to compile and install Erlang:

```
./configure --prefix=/usr/local/erlang
```

```
make && make install
```

vi. After Erlang is installed, run the following command to configure the environment variable for Erlang:

```
echo 'export PATH=$PATH:/usr/local/erlang/bin' >> /etc/profile
```

vii. Run the following command to apply the configured environment variable:

```
source /etc/profile
```

viii. Run the following commands to go back to the */root* directory. View the Erlang version to check whether Erlang is installed.

```
cd
erl -version
```

A command output similar to the following one indicates that Erlang is installed.

```
[root@test ~]# erl -version
Erlang (SMP,ASYNC_THREADS,HIPE) (BEAM) emulator version 10.1
```

3. Download and install RabbitMQ.

   Different RabbitMQ versions are compatible with different Erlang versions. For more information, see RabbitMQ Erlang Version Requirements. In this example, Erlang 21.1 is used, and you must download RabbitMQ 3.7.8.

   i. Run the following command to download the RabbitMQ installation package:

   ```
   wget https://github.com/rabbitmq/rabbitmq-server/releases/download/v3.7.8/rabbitmq-server-generic-unix-3.7.8.tar.xz
   ```

   ii. Run the following command to decompress the RabbitMQ installation package:

   ```
   tar -xvf rabbitmq-server-generic-unix-3.7.8.tar.xz
   ```

   iii. After the package is decompressed, run the following command to configure the environment variable for RabbitMQ:

   ```
   echo 'export PATH=$PATH:/root/rabbitmq_server-3.7.8/sbin' >> /etc/profile
   ```

   iv. Run the following command to apply the configured environment variable:

   ```
   source /etc/profile
   ```

4. Configure RabbitMQ.

   i. Run the following command to start RabbitMQ and run RabbitMQ in the background:

   ```
   rabbitmq-server -detached
   ```

   > **Notice** This command starts RabbitMQ only in the current runtime environment. If the instance restarts, RabbitMQ does not automatically start. We recommend that you use Cloud Assistant to configure RabbitMQ to automatically start on instance startup. For more information, see Use Cloud Assistant to Configure RabbitMQ to automatically start on instance startup.

   ii. Run the following command to enable the RabbitMQ monitoring plug-in:

   ```
   rabbitmq-plugins enable rabbitmq_management
   ```

   To disable the RabbitMQ monitoring plug-in, you can run the `rabbitmq-plugins disable rabbitmq_management` command.

   iii. To ensure data security, we recommend that you run the following command to delete the default user of RabbitMQ.

   The default username and password of RabbitMQ are both `guest`.

   ```
   rabbitmqctl delete_user guest
   ```

iv. Create a RabbitMQ administrator user.

a. Run the following command to create a user:

```
rabbitmqctl add_user <Username> <Password>
```

Specify the *<Username>* and *<Password>* parameters.

b. Run the following command to set the new user as an administrator:

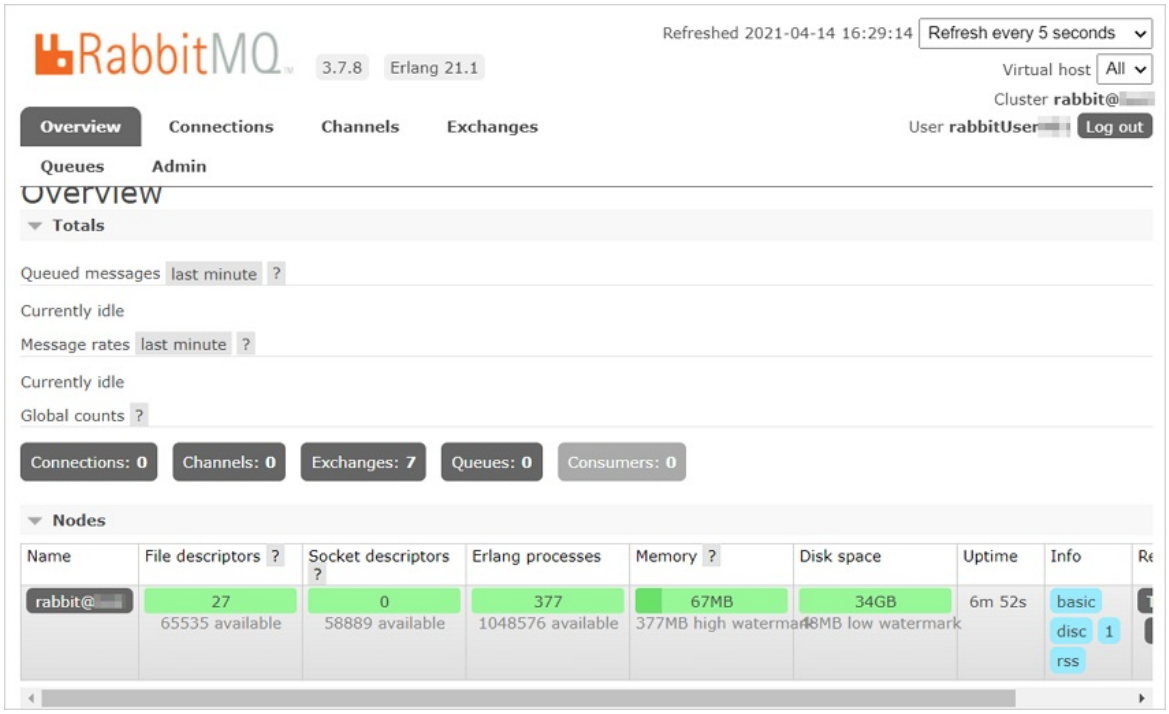
```
rabbitmqctl set_user_tags <Username> administrator
```

c. Run the following command to grant all permissions to the new user:

```
rabbitmqctl set_permissions -p / <Username> ".*" ".*" ".*"
```

5. Access `<Public IP address of the Linux instance>:15672` in a browser on your computer. The following page indicates that RabbitMQ is installed.

6. Enter the username and password of the RabbitMQ administrator user that you created and click **Login** to access the RabbitMQ management interface.
The RabbitMQ management interface appears, as shown in the following figure.



Use Cloud Assistant to Configure RabbitMQ to automatically start on instance startup

1. Configure the *rabbitmq-server* file for RabbitMQ.

i. Run the following command to edit the *rabbitmq-server* file:

```
vim /root/rabbitmq_server-3.7.8/sbin/rabbitmq-server
```

ii. Press *shift+:* and enter `set nu` to view line numbers of the file.

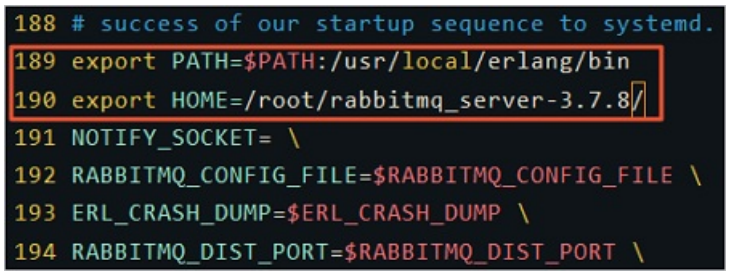
iii. Press *shift+:* and enter `189` to go to Line 189.

iv. Press the /key to enter the edit mode.

Add the following content to Line 189:

```
export PATH=$PATH:/usr/local/erlang/bin
export HOME=/root/rabbitmq_server-3.7.8/
```

The following figure shows the lines after the content is added.



v. Press the *Esc* key, enter `:wq`, and then press the Enter key to save and close the file.

2. Call the ECS API RunCommand operation to configure RabbitMQ to automatically start on instance startup.

For more information, see Quick start. To configure RabbitMQ to automatically start on instance startup, you must configure the parameters listed in the following table.

Parameter	Description	Valid or example value
RegionId	The region ID of the instance.	Example: `cn-hangzhou`.
Name	The name of the Cloud Assistant command.	Example: `start-rabbitmq`.
Type	The language type of the command.	Set the value to `RunShellScript`.
CommandContent	The plaintext content of the command.	Set the value to `/root/rabbitmq_server-3.7.8/sbin/rabbitmq-server -detached`. **Note** The command is used to start RabbitMQ.
RepeatMode	Specifies how to run the command.	Set the value to `EveryReboot`.
InstanceId.N	The ID of the instance where RabbitMQ is deployed.	Example: `i-bp12f1b0i3r7adm3****`.

The following code shows a sample success response in the JSON format. Subsequently, the command used to start RabbitMQ is triggered every time you restart the instance.

```
{
  "RequestId": "8B4BFE47-F1E3-48D1-B405-CA783B697046",
  "CommandId": "c-hz01gvo1ri9****",
  "InvokeId": "t-hz01gvo1rig****"
}
```

5.9. Deploy and use SVN

5.9.1. Overview

Apache Subversion (SVN) is an open source version control system that manages ever-changing data. This topic describes the terms and operations related to SVN.

SVN

The data that SVN manages is stored in a repository. This repository records all changes of files so that you can reverse the data to an earlier version or review the change history of files. The following section describes the terms and operations related to SVN:

- Repository: stores source code.
- Checkout: checks out source code to a local directory.
- Commit: commits modified code to the repository.
- Update: synchronizes source code in the repository to a local directory.

You can perform the following steps to manage code in SVN:

1. Checkout: Check out source code to a local directory.
2. Other users modify and commit the source code to the repository.
3. Update: Obtain the updates of the source code from the repository.
4. Modify and debug the source code.
5. Commit: Commit the debugged source code to the repository, so other users can view your modifications.

SVN manages source code by line. When you and other users modify the code in a file at the same time:

- If the modified code is in different lines, SVN automatically merges the modifications.
- If the modified code is in the same line, SVN prompts a file conflict. You must manually confirm the modification to resolve the conflict.

Procedure

SVN supports access over HTTP or based on svnserve. You can deploy these access methods. For more information, see the following topics:

- Deploy SVN by using svnserve
- Deploy SVN over HTTP

After you deploy SVN, you can commit modifications, obtain updates, and restore files by using SVN. For more information, see Use SVN.

5.9.2. Deploy SVN by using svnserve

This topic describes how to use svnserve to deploy Apache Subversion (SVN).

Prerequisites

-
- An Elastic Compute Service (ECS) instance of the ecs.g6.large instance type is created and runs a CentOS operating system. For information about how to create an ECS instance, see Creation method overview.
- Inbound rules are added to the security groups of the instance to allow traffic on port 3690, which is the default port of SVN. For more information, see Add a security group rule.

Context

In this topic, the following software versions are used to manually deploy SVN. Operations may vary based on your software versions.

- Operating system: CentOS 7.2 64-bit public image
- SVN: 1.7.14

Procedure

- Step 1: Install SVN
- Step 2: Configure SVN
- Step 3: Use a Windows client to test SVN

Step 1: Install SVN

1. Connect to a Linux instance by using a password.
2. Run the following command to install SVN:

   ```
   yum install subversion
   ```

3. Run the following command to check the SVN version:

   ```
   svnserve --version
   ```

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

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

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

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

   Cyrus SASL authentication is available.
   ```

Step 2: Configure SVN

1. Run the following command to create a root directory for an SVN repository:

```
mkdir /var/svn
```

2. Run the following commands in sequence to create an SVN repository.

```
cd /var/svn
```

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

3. Run the following commands in sequence to check the files that are automatically generated in the SVN repository:

```
cd svnrepos
```

```
ls
```

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

The following table describes the SVN directories and files.

Directory and file	Description
db	Stores all version control data files.
hooks	Stores hook scripts.
locks	The client used to track access to the SVN repository.
format	The text file that contains a single integer value. The value indicates the version number of the current SVN repository.
conf	The configuration file of the SVN repository, which stores the usernames and permissions for accessing the repository.

4. Configure the username and password of the SVN repository.
   i. Run the `cd conf/` command.
   ii. Run the `vi passwd` command to open the configuration file.
   iii. Press the `I` key to enter the edit mode.

iv. Move the pointer over `[users]` and add the username and password.

> **Note** Add the username and password in the following format: username = password. Example: userTest = passWDTest, as shown in the following figure. You must add a space before and after the equal sign (=).

```
### This file is an example password file for svnserve.
### Its format is similar to that of svnserve.conf. As shown in the
### example below it contains one section labelled [users].
### The name and password for each user follow, one account per line.

[users]
# harry = harryssecret
# sally = sallyssecret
userTest = passWDTest
```

v. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.

5. Configure the read and write permissions for the account.
   i. Run the `vi authz` command to open the access control file.
   ii. Press the `I` key to enter the edit mode.
   iii. Move the pointer over the end of the file and add the following code. In the code, userTest specifies the username, r specifies the read permissions, and w specifies the write permissions.

```
[/]
userTest=rw
```

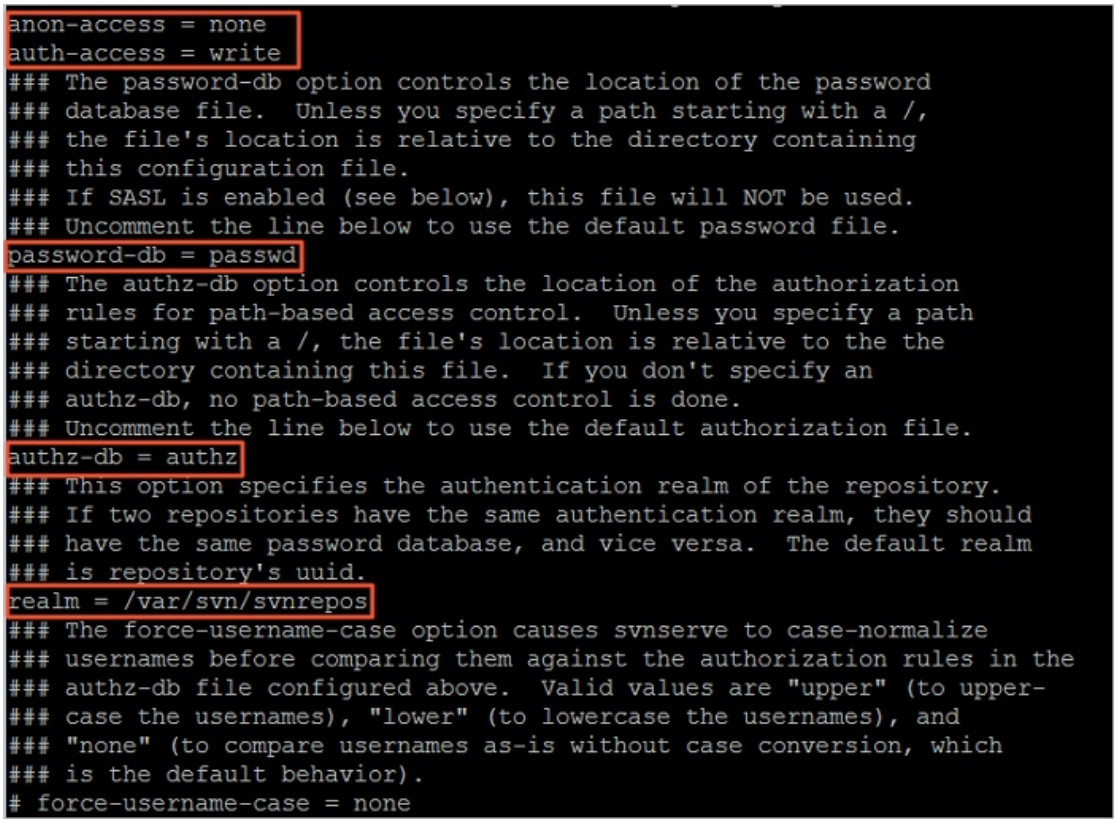
```
# [repository:/baz/fuz]
# @harry_and_sally = rw
# * = r
[/]
userTest=rw
```

   iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.
6. Modify the configurations of SVN.
   i. Run the `vi svnserve.conf` command to open the configuration file of SVN.
   ii. Press the `I` key to enter the edit mode.

iii. Move the pointer over the following lines and delete the number sign (#) and space from the beginning of each line.

> **Note** Lines cannot start with a space. You must add a space before and after the equal sign (=).

```
anon-access = read # Grant read permissions to anonymous users. You can also set an
on-access to none to deny access by anonymous users. If you set anon-access to none
, the operation dates in the SVN log can be shown properly.
auth-access = write # Grant write permissions.
password-db = passwd # Specify the password database file.
authz-db = authz # Specify the file that stores the authorization rules for path-ba
sed access control.
realm = /var/svn/svnrepos # Specify the authorization realm of the SVN repository.
```



iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.

7. Run the following command to start the SVN repository:

   The absolute path to the SVN repository is specified in the following example command:

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

   > **Note** You can run the `killall svnserve` command to stop SVN.

8. Run the `ps -ef |grep svn` command to check whether SVN is started.

A command output similar to the following one indicates that SVN is started.

```
[root@test conf]# ps -ef |grep svn
root     13030     1  0 10:42 ?        00:00:00 svnserve -d -r /var/svn/svnrepos/
root     13140  9283  0 10:43 pts/0    00:00:00 grep --color=auto svn
```

Step 3: Use a Windows client to test SVN

1. Download and install the TortoiseSVN client on your computer.
2. Right-click the blank area in the on-premises project folder.

   In this example, the project folder is *C:\Test*.
3. Select **SVN Checkout...** from the shortcut menu.
4. Configure the following settings and click **OK**:
   - In the **URL of repository:** field, enter the URL of the SVN repository from which to check out a working copy. In these examples, SVN is started by using the `svnrepos` repository and svnserve works only for the svnrepos repository. The URL of the SVN repository is in the format of *svn://<Public IP address of the instance>*.

     > **Note** If SVN is started in the upper directory of the SVN repository, the name of the SVN repository must be added to the URL that is checked out from SVN.

   - Set the **Checkout directory:** field. In this example, the *C:\Test* directory is used.

     > **Note** The first time you log on to SVN, you must enter the username and password that you have configured in the *passwd* file.

5.9.3. Deploy SVN over HTTP

This topic describes how to deploy Apache Subversion (SVN) over HTTP.

Prerequisites

-
-
-
- An instance that runs a CentOS operating system is created. For more information, see Creation method overview.
- Inbound rules are added to the security groups of the instance to allow traffic on port 3690, which is the default port of SVN. For more information, see Add a security group rule.

Context

In this topic, the following instance configurations and software versions are used to manually deploy SVN. The procedure may vary based on your actual configurations.

- Instance type: ecs.c6.large
- Operating system: CentOS 7.2 64-bit public image
- SVN: 1.7.14
- Apache HTTP Server: 2.4.6

You can also use Alibaba Cloud Marketplace images to deploy SVN. For example, you can use SVN images provided by Alibaba Cloud Marketplace to deploy SVN. For more information, see the "User guide" section on the SVN version control (CentOS 64-bit) page.

Step 1: Install SVN

1. Connect to the Linux instance. For more information, see Connect to a Linux instance by using a password.
2. Run the following command to install SVN:

```
yum install subversion -y
```

3. Run the following command to check the SVN version:

```
svnserve --version
```

Step 2: Install Apache

1. Run the following command to install httpd:

```
yum install httpd -y
```

2. Run the following command to check the httpd version:

```
httpd -version
```

Step 3: Install mod_dav_svn

Run the following command to install mod_dav_svn:

```
yum install mod_dav_svn -y
```

Step 4: Configure SVN

1. Run the following commands in sequence to create an SVN repository:

```
mkdir /var/svn
```

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

2. Run the following command to change the user group of the SVN repository to apache:

```
chown -R apache:apache /var/svn/svnrepos
```

3. Run the following commands in sequence to check the files that are automatically generated in the SVN repository:

```
cd svnrepos
```

```
ls
```

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

The following table describes the SVN directories and files.

Directory and file	Description
db	Stores all version control data files.
hooks	Stores hook scripts.
locks	The client used to track access to the SVN repository.
format	The text file that contains a single integer value. The value indicates the version number of the current SVN repository.
conf	The configuration file of the SVN repository, which stores the usernames and permissions for accessing the repository.

4. Run the following command to add a username and password for the SVN repository.

   By default, the password for SVN is stored as plaintext. You must separately generate a passwd file for HTTP because HTTP does not support plaintext passwords. In this example, the added username is `userTest` and the password is `passWDTest`. Run one of the following commands:

   - If this is the first time that you add a user for the SVN repository, run the following command that contains `-c` to generate the passwd file:

     ```
     htpasswd -c /var/svn/svnrepos/conf/passwd userTest
     ```

   - If this is not the first time that you add a user for the SVN repository, run the following command to generate the passwd file:

     ```
     htpasswd /var/svn/svnrepos/conf/passwd userTest
     ```

   Set the password of the user.

5. Run the following command to go to the *conf* directory:

   ```
   cd /var/svn/svnrepos/conf/
   ```

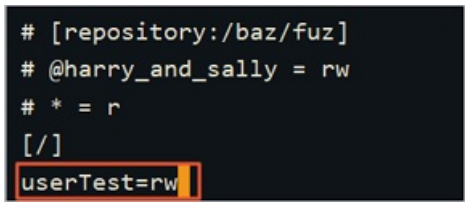
6. Configure the read and write permissions for the account.

   i. Run the `vi authz` command to open the access control file.

   ii. Press the `I` key to enter the edit mode.

   iii. Move the pointer over the end of the file and add the following code. In the code, userTest specifies the username, r specifies the read permissions, and w specifies the write permissions.

   ```
   [/]
   userTest=rw
   ```

   iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.

7. Modify the configurations of SVN.

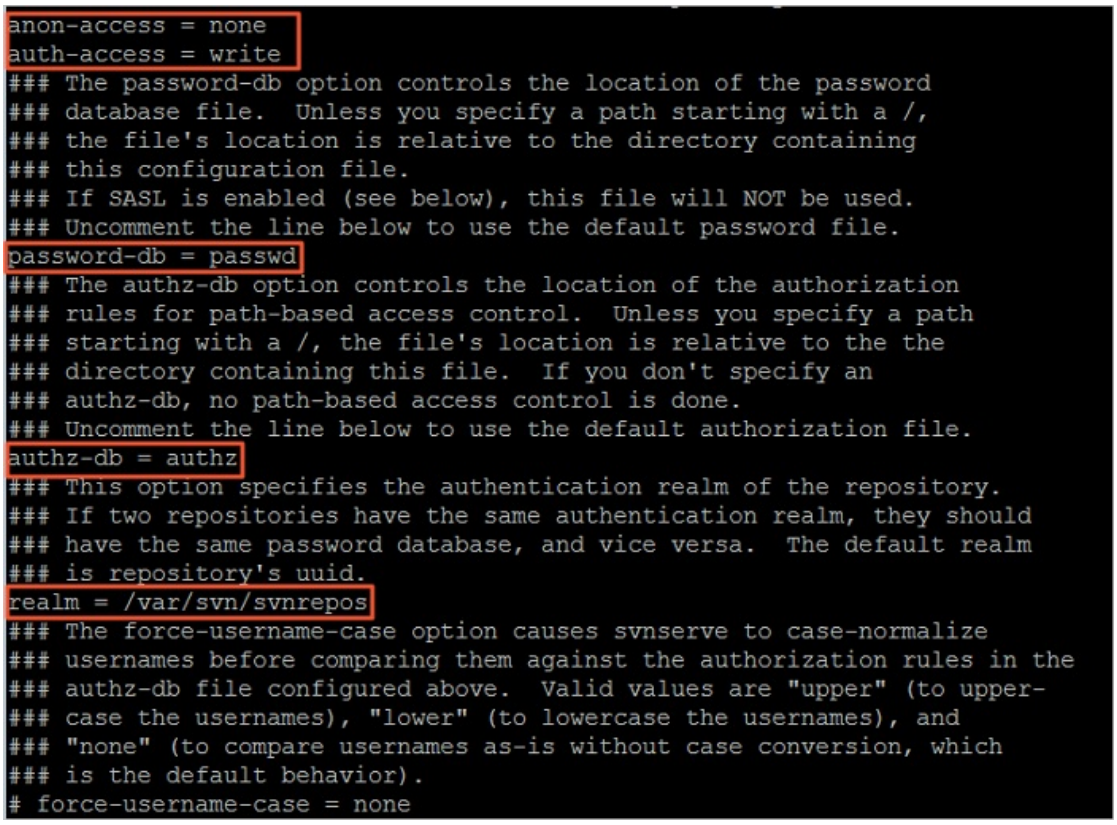
   i. Run the `vi svnserve.conf` command to open the configuration file of SVN.

ii. Press the `I` key to enter the edit mode.

iii. Move the pointer over the following lines and delete the number sign (#) and space from the beginning of each line.

> **Note** Lines cannot start with a space. You must add a space before and after the equal sign (=).

```
anon-access = read # Grant read permissions to anonymous users. You can also set an
on-access to none to deny access by anonymous users. If you set anon-access to none
, the operation dates in the SVN log can be shown properly.
auth-access = write # Grant write permissions.
password-db = passwd # Specify the password database file.
authz-db = authz # Specify the file that stores the authorization rules for path-ba
sed access control.
realm = /var/svn/svnrepos # Specify the authorization realm of the SVN repository.
```



iv. Press the `Esc` key to exit the edit mode and enter `:wq` to save and close the file.

8. Run the following command to start the SVN repository:

   The absolute path to the SVN repository is specified in the following example command:

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

   > **Note** You can run the `killall svnserve` command to stop SVN.

9. Run the `ps -ef |grep svn` command to check whether SVN is started.

A command output similar to the following one indicates that SVN is started.

```
[root@test conf]# ps -ef |grep svn
root     13030     1  0 10:42 ?        00:00:00 svnserve -d -r /var/svn/svnrepos/
root     13140  9283  0 10:43 pts/0    00:00:00 grep --color=auto svn
```

Step 5: Configure Apache

1. Run the following command to add and edit the httpd configuration file:

   ```
   vim /etc/httpd/conf.d/subversion.conf
   ```

2. Press the `I` key to enter the edit mode.
3. Enter the following configurations:

   ```
   <Location /svn>
   DAV svn
   SVNParentPath /var/svn
   AuthType Basic
   AuthName "Authorization SVN"
   AuthzSVNAccessFile /var/svn/svnrepos/conf/authz
   AuthUserFile /var/svn/svnrepos/conf/passwd
   Require valid-user
   </Location>
   ```

4. Press the `Esc` key and enter `:wq` to save and close the file.
5. Run the following command to start the Apache service:

   ```
   systemctl start httpd.service
   ```

Step 6: Use a browser to test the access to SVN

1. Open the browser in your computer.
2. In the address bar, enter a URL in the `http://<Public IP address of the ECS instance>/svn/<SVN repository name>` format and press the Enter key. In this example, the SVN repository name is *svnrepos*.
3. Enter your username and password that you configured in the *passwd* file. In this example, the username is userTest and the password is passWDTest.

   The following command output indicates that the created SVN repository is accessed.

5.9.4. Use SVN

After you deploy Apache Subversion (SVN), you can check out a project from the SVN repository to a local directory, commit local modifications to the repository, obtain updates from the repository, and reverse deleted files.

Prerequisites

You have deployed SVN. For more information, see Deploy SVN by using svnserve and Deploy SVN over HTTP.

Commit modifications

To commit local modifications to the repository, follow these steps:

1. Right-click the blank area in a project folder, and select **SVN Commit**.
2. Enter the revision comments, select the modifications that you want to commit, and then click **OK**. Then, the original project in the repository is overwritten by the project that you have committed.

> **Note** A conflict occurs when two users modify the same object of the same version and commit the modifications. In this case, one of the commitments will fail due to the backward version. To avoid this issue, you can back up your local project, check out the latest project from the repository, overwrite the latest project with your local project, and then commit the modified project.

Obtain updates

After the project in the SVN repository is updated, you can right-click a blank area in the local project folder, and select **SVN Update** to download and display all updates.

> **Note** When you right-click a blank area in the local project folder and select SVN Update, all files in the project folder are overwritten. Therefore, we recommend that you back up the original project folder before the update operation, in case some required content may be overwritten.

Reverse deleted data

To reverse deleted data, follow these steps:

1. Open a local project folder, right-click the blank area in the folder, and then select **SVN Checkout** to check out data.
2. Delete the data you checked out.
3. Choose between the following methods to reverse the deleted data based on your commitment conditions.
   - If you have not committed the delete operation, right-click the blank area in the folder, and choose **TortoiseSVN > SVN Revert**.
   - If you have committed the delete operation, the modification has been synchronized to the repository, and the corresponding data has also been deleted from the repository. Therefore, to reverse the deleted data, follow these steps:
     a. Check the revision history and determine the data that has been deleted.
     b. Right-click the deleted data and select Revert to this revision.
4. Open the original project folder, right-click the reversed data, and then select **SVN Commit** to synchronize the local reversed data to the repository.

6.Use the Vim editor

Vim is a text editor that is developed as an improved version of the vi editor. It can display text with extra format details, such as font color and underline. Vim is an essential tool in Linux. For example, you can use this tool to edit configuration files of web applications. This topic describes the modes and commonly used commands of Vim.

Context

The following table describes the different modes of Vim.

Mode	Description	Mode switching method
Normal mode	In this mode, you can copy, paste, and delete characters or lines.	• Vim enters the normal mode when you run the `vim <File name>` command to open a file. • To switch from other modes to this mode, press the `Esc` key.
Insert mode	In this mode, you can insert characters.	To switch from the normal mode to this mode, enter one of the following characters: `i, I, a, A, o, O`. **Note** `-- INSERT --` are shown in the lower-left corner of the editor after Vim enters the insert mode.
Replace mode	In this mode, you can replace characters.	To switch from the normal mode to this mode, enter `R`. **Note** `-- REPLACE --` is shown in the lower-left corner of the editor after Vim enters the replace mode.
Visual mode	In this mode, you can select a range of text. You must select a range of text before you run commands such as copy, replace, and delete on the selected text.	To switch from the normal mode to this mode, enter `v`. **Note** `-- VISUAL --` is shown in the lower-left corner of the editor after Vim enters the visual mode.
Command mode	In this mode, you can find and replace strings, have line numbers displayed, save file changes, and exit the editor.	To switch from the normal mode to this mode, enter `:`.

Vim supports the following commands:

- Insert

- Replace
- Delete

Insert

Basic commands:

- i: inserts a character to the left of the current character.
- I: inserts a character at the start of the current line.
- a: inserts a character to the right of the current character.
- A: inserts a character at the end of the current line.
- o: inserts a new line below the current line.
- O: inserts a new line above the current line.

Assume that you want to edit an *example.conf* file that contains the following content:

```
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf
```

- Example 1: Insert `Location` at the first line of the *example.conf* file. Perform the following operations:
  i. Run the `vim example.conf` command to open the file in normal mode.
  ii. Enter `i` to switch to the insert mode.
  iii. Enter `Location`.
  iv. Press the Enter key to switch to a new line.
  v. Press the `Esc` key to exit the insert mode.
  vi. Enter `:wq` to save the changes to the file and then exit the editor.

  After the specified content is inserted to the *example.conf* file, the file contains the following content:

  ```
  Location
  # To be able to use the functionality of a module which was built as a DSO you
  # have to place corresponding `LoadModule' lines at this location so the
  # directives contained in it are actually available _before_ they are used.
  # Statically compiled modules (those listed by `httpd -l') do not need
  # to be loaded here.
  #
  # Example:
  # LoadModule foo_module modules/mod_foo.so
  #
  Include conf.modules.d/*.conf
  ```

- Example 2: Insert `#` at the start of line 10 in the *example. conf* file. Perform the following operations:
  i. Run the `vim example.conf` command to open the file in normal mode.
  ii. Enter `:10` to move the cursor to line 10.
  iii. Enter `I` to switch to the insert mode.
  iv. Enter `#`.
  v. Press the `Esc` key to exit the insert mode.
  vi. Enter `:wq` to save the changes to the file and then exit the editor.

  After the specified content is inserted to the *example.conf* file, the file contains the following content:

```
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
#Include conf.modules.d/*.conf
```

- Example 3: Insert `LoadModule rewrite_module modules/mod_rewrite.so` in the line below the `Include conf.modules.d/*.conf` line of the *example.conf* file. Perform the following operations:
  i. Run the `vim example.conf` command to open the file in normal mode.
  ii. Run the `/Include conf.modules.d/*.conf` command to find the line on which you want to perform the insert operation.
  iii. Enter `o` to switch to the insert mode.
  iv. Enter `LoadModule rewrite_module modules/mod_rewrite.so`.
  v. Press the `Esc` key to exit the insert mode.
  vi. Enter `:wq` to save the changes to the file and then exit the editor.

  After the specified content is inserted to the *example.conf* file, the file contains the following content:

```
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf
LoadModule rewrite_module modules/mod_rewrite.so
```

Replace

Basic commands:

R: replaces the highlighted characters, until you press the `Esc` key to exit the replace mode.

Assume that you want to edit an *example.conf* file that contains the following content:

```
# 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
```

Example: To replace `AllowOverride None` with `AllowOverride All` in the *example.conf* file, perform the following operations:

1. Run the `vim example.conf` command to open the file in normal mode.
2. Run the `/AllowOverride None` command to find the line on which you want to perform the replace operation.
3. Move the cursor to the first letter of `None`.
4. Enter `R` to switch to the replace mode.
5. Enter `All` and a space.

   > **Note** The word `None` has four characters, but the word `All` has three characters. To replace all four characters in None, enter an extra space after the three characters in All.

6. Press the `Esc` key to exit the replace mode.
7. Enter `:wq` to save the changes to the file and then exit the editor.

   The replaced *example.conf* file contains the following content:

   ```
   # 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 All
   ```

Delete

Basic commands:

- x: deletes the highlighted character.
- nx (n represents a number): deletes the highlighted character and the n-1 characters after it.
- dd: deletes the line in which the cursor is located.
- ndd (n represents a number): deletes the line in which the cursor is located and the n-1 lines below it.

Assume that you want to edit an *example.conf* file that contains the following contents:

```
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80
```

- Example 1: Delete `#` at the start of the `#Listen 12.34.56.78:80` line of the *example.conf* file. Perform the following operations:
  i. Run the `vim example.conf` command to open the file in normal mode.
  ii. Run the `/#Listen 12.34.56.78:80` command to find the line on which you want to perform the delete operation so that the cursor is on the `#` character.
  iii. Enter `x` to delete `#`.
  iv. Enter `:wq` to save the changes to the file and then exit the editor.

  After the specified content is deleted from the *example.conf* file, the file contains the following content:

  ```
  # Listen: Allows you to bind Apache to specific IP addresses and/or
  # ports, instead of the default. See also the <VirtualHost>
  # directive.
  #
  # Change this to Listen on specific IP addresses as shown below to
  # prevent Apache from glomming onto all bound IP addresses.
  #
  Listen 12.34.56.78:80
  Listen 80
  ```

- Example 2: Delete the `#Listen 12.34.56.78:80` line and the line below it in the *example.conf* file. Perform the following operations:
  i. Run the `vim example.conf` command to open the file in normal mode.
  ii. Run the `/#Listen 12.34.56.78:80` command to find the lines on which you want to perform the delete operation.
  iii. Enter `2dd` to delete the following contents.

  ```
  #Listen 12.34.56.78:80
  Listen 80
  ```

  iv. Enter `:wq` to save the changes to the file and then exit the editor.

  After the specified content is deleted from the *example.conf* file, the file contains the following content:

```
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
```