Alibaba Cloud
Elastic Compute Service

Images

Issue: 20190418

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 due to product version upgrades , adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall 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 the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults " and "as available" basis. 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 bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, 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 content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, 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 the Alibaba Cloud website, product programs, or content 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 contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information 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.
	This warning information 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 restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus , page names, and other UI elements.	Click OK.
`Courier font`	It is used for commands.	Run the `cd /d C:/windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*

Style	Description	Example
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich {stand | slave}`

Contents

1 Images

An image is a running environment template for ECS instances. It generally includes an operating system and pre-installed software. An image works as a file copy that includes data from one or more disks. These disks can be a single system disk, or a combination of the system disk and data disks.

Image overview

ECS provides a range of image types that you can use to easily access image resources.

Image type	Type description	Technical support
Public image	Alibaba Cloud provides the following types of public images: · Aliyun Linux image: The Aliyun Linux image is a custom, native operating system provided by Alibaba Cloud for ECS. The Aliyun Linux image has undergone stringent testing to guarantee its security, stability, and normal startup and operation. · Third-party commercial image and open-source licensed image: Such images include Windows Server, Ubuntu, CentOS, Red Hat Enterprise Linux, Debian, SUSE Linux, FreeBSD and CoreOS. These images have undergone stringent testing conducted by Alibaba Cloud to guarantee their security, stability, and normal startup and operation.	· Aliyun Linux image : Alibaba Cloud provides *technical support*. · Third-party commercial image and open-source licensed image: You can contact the OS vendors or open source communities for technical support. Additionally, Alibaba Cloud provides technical support to assist investigation into various image-related and system-related problems.

Image type	Type description	Technical support
Custom image	Custom images include imported images, and custom images created from public images and Alibaba Cloud Marketplace images.	You can contact the OS vendors for technical support. Additionally, Alibaba Cloud provides technical support in problem investigation.
Marketplace image	*Marketplace* images are licensed and provided by Independent Software Vendors (ISVs) and sold in Alibaba Cloud Marketplace. Such images have undergone stringent testing conducted by the respective ISVs and Alibaba Cloud to guarantee security. Marketplace images provide not only the OSs required for applications, but also the configuration environments. This eliminates complicated installation and configuration process and helps you deploy ECS in one click.	You can contact the image vendors for technical support.

Public images

Public images are fully licensed to provide a highly stable operating environment. You can customize your application environment based on a public image. Different instance types correspond to different available public images. For information about the built-in services of public images releases, go to the official website of the OS vendors.

Alibaba Cloud regularly releases and updates public images. For more information, see *Image release notes*. You can view available public images in the *Public image list in the ECS console*. The following table lists the ECS public images.

OS type	OS version	OS type	OS version
Windows Server	· Windows Server 2008 Standard Edition SP2 32-bit · Windows Server 2008 R2 Enterprise Edition 64-bit · Windows Server 2008 R2 Enterprise Edition 64-bit · Windows Server 2012 R2 Datacenter 64-bit · Windows Server 2012 R2 Datacenter 64-bit · Windows Server 2016 R2 Datacenter 64-bit · Windows Server 2016 R2 Datacenter 64-bit · Windows Server Version 1809 Datacenter 64-bit · Windows Server Version 1809 Datacenter 64-bit	CentOS	· CentOS 6.8 64-bit · CentOS 6.8 32-bit · CentOS 6.9 64-bit · CentOS 7.2 64-bit · CentOS 7.3 64-bit · CentOS 7.4 64-bit · CentOS 7.5 64-bit · CentOS 7.6 64-bit
SUSE Linux	· SUSE Linux Enterprise Server 11 SP4 64-bit · SUSE Linux Enterprise Server 12 SP4 64-bit	Debian	· Debian 8.9 64-bit · Debian 9.6 64-bit
Red Hat	· Red Hat Enterprise Linux 7.5 64-bit · Red Hat Enterprise Linux 7.4 64-bit · Red Hat Enterprise Linux 6.9 64-bit	Ubuntu	· Ubuntu 14.04 64-bit · Ubuntu 16.04 64-bit · Ubuntu 18.04 64-bit
FreeBSD	FreeBSD 11.1 64-bit	OpenSUSE	OpenSUSE 42.3 64-bit
Aliyun Linux	Aliyun Linux 17.1 64-bit	CoreOS	CoreOS 1745.7.0 64bit

Custom images

After you successfully create or import a custom image, the status of the image becomes `Available`. You can then use this image to create an instance, share the image with other Alibaba Cloud accounts, or copy the image to other Alibaba Cloud

regions under your account. The following figure shows the typical usage of a custom image.

You can create a custom image by using the following methods:

- *Create a custom image by using a snapshot*
- *Create a custom image by using an instance*
- *Import an on-premises custom image*

> Note:
> A custom image must be imported in one of the following formats: VHD, qcow2, or RAW. If the custom image is in another format, it must be converted before it can run in ECS. For more information, see *Convert image file format*.

After creating custom images, you can perform the following operations:

- *Replace the OS of an instance*
- *Copy your custom images from one region to another*
- *Share your custom images with other Alibaba Cloud accounts*
- *Export your custom images to on-premises testing environments or private cloud environments*
- *Manage your custom images*

Billing details

We recommend that you maintain a sufficient balance in the linked credit card or PayPal account to complete the payment or preauthorization. For more information, see *Pricing overview*.The ECS image billing details are as follows:

Image type	Billing description
Public image	Only Windows Server and Red Hat Enterprise Linux public images incur fees, which are included in the bill when an instance is created. Windows Sever public images or Red Hat Enterprise Linux public images are provided with certified licenses and authorized support from Microsoft or Red Hat, respectively. · Red Hat Enterprise Linux: Billing is related to the instance type. · Windows Server: Free of charge in Alibaba Cloud regions in Mainland China. For other countries and regions, the use of Windows Server public images incur fees. Other public images are available free of charge.

Image type	Billing description
Custom image	· If you *Create a custom image by using a snapshot*: - If the image used by the system disk snapshot is from the Alibaba Cloud Marketplace, the bill may include fees for the image and fees for the snapshot capacity. - If the image used by the system disk snapshot is not from the Alibaba Cloud Marketplace, the bill may include fees for the snapshot capacity. · If you *Create a custom image by using an instance*, and the image used by the instance is from the Alibaba Cloud Marketplace, the billing policies of the Alibaba Cloud Marketplace and the ISV apply.
Marketplace image	The billing policies of the ISVs apply.
Shared image	If a shared image comes from the Alibaba Cloud Marketplace, the billing policies of the ISV apply.

Related operations

Console operations

- You can *Create an instance from a custom image*.
- You can change the system disk of an ECS instance by using either of the following methods:
  - *Replace the image of the system disk with a public image*.
  - *Replace the image of the system disk with a non-public image*.
- You can obtain a custom image by using the following methods:
  - *Create a custom image by using a snapshot*.
  - *Create a custom image by using an instance*.
  - *Import an on-premises custom image*.
- After creating custom images, you can perform the following operations:
  - *Copy your custom images from one region to another*.
  - *Share your custom images with other Alibaba Cloud accounts*.
  - *Export your custom images to on-premises testing environments or private cloud environments*.

API operations

For more information, see the *APIs for images*.

2 Public image

2.1 Release notes

This topic describes the release notes of images and relevant updates.

March 27, 2019

Release	Description
Aliyun Linux 2	· Image ID: aliyun-2.1903-x64-20G-alibase-20190327.vhd · Kernel version: 4.19.24-9.al7.x86_64 · Released in: China North 2, China North 3, China North 5, and Dubai · What's new: released Aliyun Linux 2

March 19, 2019

Release	Description
CoreOS 2023.4.0	· Image ID: coreos_2023_4_0_64_30G_alibase_20190319.vhd · Kernel version: 4.19.25-coreos · Released in: all regions · What's new: updated to the latest system patches

March 18, 2019

Release	Description
Windows Server 2019 Datacenter Edition	· Image ID: - win2019_64_dtc_1809_zh-cn_40G_alibase_20190318.vhd (Chinese edition) - win2019_64_dtc_1809_en-us_40G_alibase_20190318.vhd (English edition) · Released in: China North 1, China East 2, and China South 1 · What's new: new release

Release	Description
Windows Server 2016 Datacenter Edition	· Image ID: - win2016_64_dtc_1607_zh-cn_40G_alibase_20190318.vhd (Chinese edition) - win2016_64_dtc_1607_en-us_40G_alibase_20190318.vhd (English edition) · Released in: all regions · What's new: updated to the latest system patches
Windows Server 2012 R2 Datacenter Edition	· Image ID: - win2012r2_64_dtc_9600_zh-cn_40G_alibase_20190318.vhd (Chinese edition) - win2012r2_64_dtc_9600_en-us_40G_alibase_20190318.vhd (English edition) · Released in: all regions · What's new: updated to the latest system patches
Windows Server 2008 R2 Enterprise Edition	· Image ID: - win2008r2_64_ent_sp1_zh-cn_40G_alibase_20190318.vhd (Chinese edition) - win2008r2_64_ent_sp1_en-us_40G_alibase_20190318.vhd (English edition) · Released in: all regions · What's new: updated to the latest system patches
Windows Server Version 1809 Datacenter Edition	· Image ID: - winsvr_64_dtcC_1809_zh-cn_40G_alibase_20190318.vhd (Chinese edition) - winsvr_64_dtcC_1809_en-us_40G_alibase_20190318.vhd (English edition) · Released in: all regions · What's new: updated to the latest system patches

March 11, 2019

Release	Description
Debian 8.11	· Image ID: debian_8.11_64_20G_alibase_20190311.vhd · Kernel version: 3.16.0-7-amd64 · Released in: all regions · What's new: - updated to the latest system patches - Fixed invalid apt source configurations in Debian 8.9

March 1, 2019

Release	Description
Ubuntu 16.04	· Image ID: ubuntu_16_04_64_20G_alibase_20190301.vhd · Kernel version: 4.4.0-142-generic · Released in: all regions · What's new: updated to the latest system patches

February 25, 2019

Release	Description
Debian 9.8	· Image ID: debian_9_08_64_20G_alibase_20190225.vhd · Kernel version: 4.9.0-8-amd64 · Released in: China North 2, China North 3, and China North 5 · What's new: updated to the latest system patches

February 23, 2019

Release	Description
Ubuntu 18.04	· Image ID: ubuntu_18_04_64_20G_alibase_20190223.vhd · Kernel version: 4.15.0-45-generic · Released in: all regions · What's new: updated to the latest system patches

February 18, 2019

Release	Description
CentOS 7.6	· Image ID: centos_7_06_64_20G_alibase_20190218.vhd · Kernel version: 3.10.0-957.5.1.el7.x86_64 · Released in: all regions · What's new: updated to the latest system patches

January 3, 2019

Release	Description
Debian9.6	· Image ID: debian_9_06_64_20G_alibase_20190103.vhd · Kernel version: 4.9.0-8-amd64 · Released in: all regions · What's new: Enabled the systemd-networkd service.

December 22, 2018

Release	Description
Windows Server version 1809 Datacenter Edition	· Image ID: - winsvr_64_dtcC_1809_zh-cn_40G_alibase_20181222.vhd (Chinese version) - winsvr_64_dtcC_1809_en-us_40G_alibase_20181222.vhd (English version) · Released in: all regions · What's new: - Updated the image to the latest patch KB4483235 (released in December 2018). - Used Sysprep tool to generalize the image.
Windows Server 2008 R2 Enterprise Edition	· Image ID: win2008r2_64_ent_sp1_en-us_40G_alibase_20181222.vhd (English version) · Released in: all regions · What's new: - Updated the image to the latest patch KB3371318 (released in December 2018). As a result, Windows clients need to be updated with the latest patches to establish RDP connections. - Upgraded NET Framework to 4.7.2. - Used Sysprep tool to generalize the image.

December 20, 2018

Release	Description
Windows Server 2008 R2 Enterprise Edition	· Image ID: win2008r2_64_ent_sp1_zh-cn_40G_alibase_20181220.vhd (Chinese version) · Released in: all regions · What's new: - Updated the image to the latest patch KB4471318 (released in December 2018). As a result, Windows clients need to be updated with the latest patches to establish RDP connections. - Upgraded NET Framework to 4.7.2. - Used Sysprep tool to generalize the image.
Windows Server 2012 R2 Datacenter Edition	· Image ID: - win2012r2_64_dtc_9600_zh-cn_40G_alibase_20181220.vhd (Chinese version) - win2012r2_64_dtc_9600_en-us_40G_alibase_20181220.vhd (English version) · Released in: all regions · What's new: - Updated the image to the latest patch KB4471320 (released in December 2018). As a result, Windows clients need to be updated with the latest patches to establish RDP connections. - Upgraded NET Framework to 4.7.2. - Used Sysprep tool to generalize the image.
Windows Server 2016 Datacenter Edition	· Image ID: - win2016_64_dtc_1607_zh-cn_40G_alibase_20181220.vhd (Chinese version) - win2016_64_dtc_1607_en-us_40G_alibase_20181220.vhd (English version) · Released in: all regions · What's new: - Updated the image to the latest patch KB4471321 (released in December 2018). As a result, Windows clients need to be updated with the latest patches to establish RDP connections. - Upgraded NET Framework to 4.7.2. - Used Sysprep tool to generalize the image.

December 12, 2018

Release	Description
CentOS 7.6	· Image ID: centos_7_05_64_20G_alibase_20181212.vhd · Kernel version: 3.10.0-957.1.3.el7.x86_64 · Released in: all regions · What's new: updated to the latest system patches
Debian 9.6	· Image ID: debian_9_06_64_20G_alibase_20181212.vhd · Kernel version: 4.9.0-8-amd64 · Released in: all regions · What's new: - Updated to the latest system patches. - Updated the cloud-init version. - Enabled the chrony service (time synchronization). - Set GRUB_TIMEOUT=1. · Known issues: *Classic network configuration issues*
Ubuntu 18.04	· Image ID: ubuntu_18_04_64_20G_alibase_20181212.vhd · Kernel version: 4.15.0-42-generic · Released in: all regions · What's new: - Updated to the latest system patches. - Updated the cloud-init version. - Enabled the chrony service (time synchronization). - Set GRUB_TIMEOUT=1.

December 10, 2018

Release	Description
CentOS 7.5	· Image ID: centos_7_05_64_20G_alibase_20181210.vhd · Kernel version: 3.10.0-862.3.3.el7.x86_64 · Released in: all regions · What's new: - Updated to the latest system patches. - Updated the cloud-init version. - Enabled the chrony service (time synchronization). - Disabled password logon by default. - Set GRUB_TIMEOUT=1.

2.2 Aliyun Linux 2

Aliyun Linux 2 is a new generation of Aliyun Linux operating system developed by Alibaba Cloud. It is intended to provide a safer, more stable, and high-performance running environment for applications on ECS instances. You can create an instance by using the Aliyun Linux 2 public image for free.

Scope of application

- Various workloads on cloud. For example, you can run databases, data analytics , Web applications, and other workloads in a production environment on Aliyun Linux 2.
- *Various instance type families*, including ECS Bare Metal Instances. The supported instance specifications are as follows:
  - vCPU: 1 vCPU~160 vCPU
  - Memory: 0.5 GiB~3,840 GiB

   Note:
  Aliyun Linux 2 does not support instances that use the Xen virtual infrastructure or instances that use the Classic network.

Advantages

Compared with other Linux systems, Aliyun Linux 2 has the following advantages:

- Tailor-made for Alibaba Cloud ECS instances, featuring faster system startup and better runtime performance.
- Provides richer operating system features through the updated Linux kernel, user-mode software, and toolkits.
- Streamlined kernel and reduced potential security risks.
- Free to use (technical support is available).

Features

New version of the Alibaba Cloud kernel

Aliyun Linux 2 is equipped with the latest version of the Alibaba Cloud kernel. The new version of the kernel provides the following features:

- Customized based on the version 4.19.24 that has been supported by the kernel community by adding new features for cloud scenarios, improving performance, and fixing major bugs.
- Customized and optimized kernel startup parameters and system configuration parameters intended for the ECS instance environment.
- Kernel crash dumping (Kdump). You can enable or disable it without rebooting the operating system.
- Kernel Live Patching (KLP).

Software package

Aliyun Linux 2 is equipped with the *Alibaba Cloud command line tool* by default. The package updates are as follows:

- The network service is switched from `network . service` to `systemd - networkd` .
- The user mode package is compatible with CentOS 7.6.1810. That is, the user mode package of CentOS 7.6.1810 can be used on Aliyun Linux 2 directly.
- Common Vulnerabilities and Exposures (CVE) fixes were updated as of the end of March 2019.

Performance optimization

Aliyun Linux 2 increases the boot speed and improves system performance at runtime, including:

- Greatly optimized startup speed for ECS instance environments.
- Optimized multi-threaded scenarios for ECS instance environments and improved multi-threaded performance for large-scale instances.
- Significantly improved performance thanks to full link optimization for MySQL database scenarios (combined with *ESSD cloud disks*).

Get Aliyun Linux 2

Aliyun Linux 2 is officially available in the ECS console of Alibaba Cloud. You can use Aliyun Linux 2 in the following ways:

- Select Public Image and then Aliyun Linux 2 when creating an ECS instance. For more information, see *Create an instance by using the wizard*.

- Update the operating system of an existing ECS instance to Aliyun Linux 2 by replacing its system disk. For more information, see *Replace the system disk by using a public image*.

Use Aliyun Linux 2

Updated system parameters

Aliyun Linux 2 updated the following kernel configuration parameters in the configuration file `/etc/sysctl.d/50-aliyun.conf`:

- `kernel.hung_task_timeout_secs = 240`: Increases the kernel hung_task timeout seconds to avoid frequent hung_task prompts.
- `kernel.panic_on_oops = 1`: Throws the Kernel Panic exception when the kernel is experiencing an Oops error. Moreover, crash details are automatically captured if Kdump is configured.
- `kernel.watchdog_thresh = 50`: Increases the thresholds for events such as hrtimer, NMI, Soft Lockup, and Hard Lockup to avoid possible kernel false positives.
- `kernel.hardlockup_panic = 1`: Throws the Kernel Panic exception when the kernel is experiencing a Hard Lockup error. Moreover, crash details are automatically captured if Kdump is configured.

You can use the `sysctl` command to view or modify the system parameters of Aliyun Linux 2 at runtime.

Updated kernel parameters

Aliyun Linux 2 updated the following kernel parameters:

- `crashkernel=0M-2G:0M,2G-8G:192M,8G-:256M`: Reserves the memory space for the Kdump function.
- `cryptomgr.notests`: Turns off the self-check of crypto during kernel startup to speed up the startup.
- `cgroup.memory=nokmem`: Turns off the kernel memory statistics function of Memory Cgroup to avoid potential kernel instability.
- `rcupdate.rcu_cpu_stall_timeout=300`: Increases the timeout threshold of RCU CPU Stall Detector to 300 seconds to avoid kernel false positives.

Run the `cat / proc / cmdline` command to view the kernel startup parameters of Aliyun Linux 2 at runtime.

Kernel version

Aliyun Linux 2 is equipped with the 4.19.24 version of Alibaba Cloud kernel by default (the current version is kernel-4.19.24-9.al7).

You can install and switch to a v3.10 series kernel compatible with CentOS 7.6.1810 as needed, and run the following command to roll back to a v3.10 kernel:

 Note:

Replacing the kernel version may result in boot failure, so please be cautious.

```
sudo yum install -y kernel-3.10.0
sudo grub2-set-default "$(grep ^menuentry /boot/grub2/grub.cfg | grep 3.10.0 | awk -F \' '{print $2}')"
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
# Restart the system
sudo reboot
```

Kdump

Aliyun Linux 2 provides the Kdump service. Kernel errors can be captured after the service is turned on, allowing you to analyze kernel crashes.

You can turn on/off the Kdump service by using the following command:

- Run the following command to turn on the Kdump service:

```
sudo systemctl enable kdump.service
sudo systemctl restart kdump.service
```

 Note:

If the memory of the selected *instance type* is less than or equal to 2 GiB, the Kdump service cannot be used.

- Run the following command to return the memory address space reserved by the Kdump service to the operating system and turn off the Kdump service:

```
sudo sh -c 'echo 0 > /sys/kernel/kexec_crash_size'
sudo systemctl disable kdump.service
sudo systemctl stop kdump.service
```

 Note:

> After the memory address space reserved by the Kdump service is returned to the operating system, the operating system must be restarted in order to enable the Kdump service again.

Configure the network

Aliyun Linux 2 uses `systemd - networkd` to configure the network by default. When you configure the network, note the following:

- The configuration file for DHCP or static IP addresses is located in the */ etc / systemd / network /* directory.
- The command to restart the network is `sudo systemctl restart systemd - networkd` .

Get the Debuginfo package and the source code package

- Run the following command to get the Debuginfo package:

```
sudo yum install - y yum - utils
sudo debuginfo - install - y < packageNam e >
```

- Run the following command to get the source package:

```
sudo yum install - y yum - utils
sudo yumdownloa der -- source < sourcePack ageName >
```

Experimental software packages

Experimental software packages are provided by Alibaba Cloud, but not rigorously tested. Therefore, their quality is not guaranteed. Aliyun Linux 2 provides the following types of experimental packages:

- Ordinary experimental software packages
  - `Python 3 . 6`
  - `Golang 1 . 11` 和 `Golang 1 . 12`
- Experimental software packages that support SCL plugins
  - `GCC 7 . 3 . 1`
  - `GDB 8 . 0 . 1`
  - `Binutils 2 . 28`
  - `Make 4 . 2 . 1`

Install a package:

- Ordinary experimental software packages
  1. Run the following command to enable YUM store:

     ```
     sudo yum install -y alinux-release-experiment als
     ```

  2. Run the following command to install a package:

     ```
     sudo yum install -y <package name>
     ```

- Experimental software packages that support SCL plugins
  1. Run the following command to install `scl-utils`:

     ```
     sudo yum install -y scl-utils
     ```

  2. Run the following command to enable YUM store:

     ```
     sudo yum install -y alinux-release-experiment als
     ```

  3. Install the packages you need from the YUM source. The following command installs GCC, GDB, Binutils, and Make tools.

     ```
     sudo yum install -y devtoolset-7-gcc devtoolset-7
     -gdb devtoolset-7-binutils devtoolset-7-make
     ```

     After the installation is complete, you can use the newer version of GCC and related tools. The sample code is as follows:

     ```
     # To view an existing SCL, you need to specify
      the library name:
      scl -l devtoolset-7
     # Run the related SCL software:
      scl enable devtoolset-7 'gcc --version'
     ```

Technical support

Alibaba Cloud provides the following technical support for Aliyun Linux 2:

- The version support period is 3 years, and the version life cycle is ended by March 31, 2022.
- Security updates in the YUM source. You can run the `yum update` command to update to the latest version.

2.3 Known issues

This topic describes the known issues and corresponding fixes of Alibaba Cloud images for different operating systems.

Debian: Classic network configuration issues

- **Issue: IP addresses cannot be automatically assigned to classic network instances through Dynamic Host Configuration Protocol (DHCP), because the Debian system disables the systemd-networkd service by default.**
- **Image ID: debian_9_06_64_20G_alibase_20181212.vhd**
- **Fix: Run the following commands to resolve the issue.**

```
systemctl   enable   systemd - networkd
systemctl   start   systemd - networkd
```

3 Custom image

3.1 Create custom image

3.1.1 Create a custom image by using a snapshot

Custom images allow you to create multiple ECS instances with identical OS and environment data.

Custom images are based on ECS disk snapshots. You can set up identical or different configurations for ECS instances that are created from images.

You can also use an instance to create an image. For more information, see *create a custom image by using an instance*.

To enhance the security of custom images created from snapshots, see *security suggestions for Alibaba Cloud custom images*.

Note:

- Custom images cannot be used across regions.
- You can change the operating system of an instance created from a custom image, and the custom image remains usable. For details, see *change the system disk (custom image)*.
- You can upgrade the instance created from a custom image, including upgrading the CPU, memory, bandwidth, and disks.
- Custom images are created independently from the billing methods of the instances from which they were created. For example, custom images created from Subscription instances can used for creating Pay-As-You-Go instances. The converse method also applies.
- If the ECS instance used for creating a custom image expires, or the data is erased (that is, the system disk used for the snapshot expires or is released), the custom image and the ECS instances created from the custom image are not affected. However, automatic snapshots are cleared when an ECS instance is released.

Restrictions for Linux instances

- Do not load data disk information in the `/ etc / fstab` file. Otherwise, instances created using this image cannot start.
- We recommend that you `umount` all data disks before creating a custom image, and then use a snapshot to create a custom image. Otherwise, ECS instances that are created based on this custom image may not start.
- Do not upgrade the kernel or operating system version.
- Do not change the system disk partitions. The system disk only supports single root partitions.
- We recommend you check the available space of the system disk to make sure that the system has available space.
- Do not modify critical system files such as `/ sbin` , `/ bin` , `/ lib` , and so on.
- Do not modify the default logon user name root.

Procedure

1. Log on to the *ECS console*.
2. Select the region.
3. In the left-side navigation pane, click Instances.
4. Find the target instance and click its instance ID, or click Manage in the Actions column.
5. In the left-side navigation pane, click Instance Snapshots. Find the target system disk and then click Create Custom Image in the Actions column.

   The snapshot must be created from system disks. Data disks cannot be used to create custom images.

   You can also click Snapshots and Images > Snapshots, and select a snapshot created from a system disk to Create Custom Image.
6. In the Create Custom Image dialog box, complete the following:
   - Confirm the snapshot ID.
   - Enter a name and description of the custom image.
   - Optional. Check Add Data Disk Snapshot, select multiple snapshots of data disks for the image, and click Add to add a data disk.

     Note:

- We recommend that you remove sensitive data from the data disk before creating a custom image to guarantee data security.
- If the snapshot disk capacity is left blank, an empty disk is created with the default capacity of 5 GiB.
- If you select available snapshots, the disk size is the same as the size of the snapshots.

7. Click Create. Then, in the left-side navigation pane, select Snapshots and Images > Images to view the images you have created.

Linux instance image FAQ

How to `umount` a disk and delete disk table data?

If `/ dev / hda5` is attached to `/ mnt / hda5`, run any of the following three commands to detach the file system.

```
umount / dev / hda5
umount / mnt / hda5
umount / dev / hda5 / mnt / hda5
```

`/ Etc / fstab` is an important configuration file in Linux. It contains the details of mounting the file system and storage devices upon startup. If you do not want to mount a specified partition when starting the instance, delete the corresponding lines from `/ etc / fstab`. For example, you can delete the following statement to disconnect xvdb1 upon startup: `/ dev / xvdb1 / leejd ext4 defaults 0 0`.

How to determine whether a data disk is detached and a custom image can be created ?

You must make sure that the statement line for automatically attaching mounting data disk has been deleted from the fstab file.

Use the `mount` command to view the information of all mounted devices. Make sure that the execution results do not contain the information of the data disk partition.

Relevant configuration files

Before creating an image, make sure that the key configuration files listed in the following table have not been modified. Otherwise, the new instance cannot start.

Configuration file	Related to	Risks if modified
`/etc/issue*`, `/etc/*-release`, and `/etc/*_version`	System release version	Modifying /etc/issue* makes the system release version unidentifiable, which can cause instance creation failure.
`/boot/grub/menu.lst` and `/boot/grub/grub.conf`	System startup	Modifying /boot/grub/menu.lst results in kernel loading failure, which means the system cannot start.
/etc/fstab	Partitions upon startup	Modifying /etc/fstab causes partition mounting failure, which means the system cannot start.
/etc/shadow	System passwords	If this file is set to read-only, the password file cannot be edited, which means instance creation fails.
/etc/selinux/config	System security policies	Modifying /etc/selinux/config and enabling SELinux results in start failure.

3.1.2 Create a custom image by using an instance

You can create a custom image based on an ECS instance. That is, you can fully copy all its disks and pack the data into an image.

During this process, snapshots are automatically created for all disks of the instance, including the system disk and data disks. All the created snapshots compose a new custom image. The following figure details this process.

For information about creating an image from a snapshot, see *create a custom image by using a snapshot*.

Considerations

- Make sure you have deleted all confidential data in the ECS instance before creating a custom image to guarantee data security.
- During creation, do not change the status of the instance. Specifically, do not stop, start, or restart the instance.

- If your custom image contains data disks, new data disks along with the ECS instance are created together. The data on the data disk duplicates the data disk snapshot in your custom image according to the mount device.
- You can export custom images that contain data disks.
- You cannot use a custom image which contains data disks to replace the system disk.

Procedure

1. Log on to the *ECS console*.
2. Select the target region.
3. In the left-side navigation pane, click Instances.
4. Find the target instance and click More > Disk and Image > Create Custom Image.
5. Enter a name and description for the image.
6. Click Create.

The image is available after all snapshots of all disks have been created.

Additional operation

See *create a custom image by using a snapshot*.

3.1.3 Use Packer to create a custom image

This topic provides information about how to install and use Packer to create a custom image.

Prerequisites

You must have an AccessKey.

> **Note:**
> Do not use the AccessKey of your Alibaba Cloud account. Instead, *create a RAM user* and use the RAM account to create the necessary *AccessKey* to maintain account security.

Step 1. Install Packer

Go to the official *Packer download page* where you can choose required version of Packer for your operating system.

To install Packer on a Linux server

1. Connect and log on to the Linux server. If the server you want to connect to is an ECS Linux instance, see *connect to a Linux instance by using a password*.
2. Run `cd /usr/local/bin` to go to the */usr/local/bin* directory.

   Note:
   The */usr/local/bin* directory is an environment variable directory. You can install Packer to this directory or another directory that has been added to the environment variable.
3. Run `wget https://releases.hashicorp.com/packer/1.1.1/packer_1.1.1_linux_amd64.zip` to download the Packer installer. You can visit the official *download page of Packer* to download installers for other versions of Packer.
4. Run `unzip packer_1.1.1_linux_amd64.zip` to unzip the package.
5. Run `packer -v` to verify Packer's installation status. If the Packer version number is returned, you have successfully installed Packer. If error command not found is returned, Packer has not been correctly installed.

To install Packer on a Windows server

The following examples uses Windows Server 2012 64-bit:

1. Connect and log on to the Windows server. If the server you want to connect to is an ECS Windows instance, see *connect to a Windows instance*.
2. Open the official *download page of Packer* and select an appropriate Packer installer for 64-bit Windows.
3. Unzip the package to a specified directory and install Packer.
4. Define the directory for Packer in the PATH environment variable.
   a. Open the Control Panel.
   b. Select All Control Panel Items > System > Advanced System Settings.
   c. Click Environment Variable.
   d. Find Path in the system variable list.
   e. Add the Packer installation directory to the Variable Value, such as *C:\Packer* as seen in this example. Separate multiple directories with half-width semicolons (;). Click OK.

5. Run `packer . exe - v` in CMD to verify Packer's installation status. If the Packer version number is returned, you have successfully installed Packer. If error command not found prompt is returned, Packer has not been correctly installed.

Step 2. Define a Packer template

Note:

To create a custom image by using Packer, you must first create a JSON format template file. In the template, you must specify the *Alibaba Cloud Image Builder* and *Provisioner* for the custom image to be created. Packer offers a diverse range of provisioners for you to choose from when configuring the content generation mode of the custom image. In the following JSON file example, the *Shell* provisioner is used as an example to illustrate how to define a Packer template.

Create a JSON file named alicloud and paste the following content:

```
{
     " variables ": {
       " access_key ": "{{ env  ` ALICLOUD_A  CCESS_KEY `}}",
       " secret_key ": "{{ env  ` ALICLOUD_S  ECRET_KEY `}}"
     },
     " builders ": [{
       " type ":" alicloud - ecs ",
       " access_key ":"{{ user  ` access_key `}}",
       " secret_key ":"{{ user  ` secret_key `}}",
       " region ":" cn - beijing ",
       " image_name ":" packer_bas  ic ",
       " source_ima  ge ":" centos_7_0  2_64_20G_a  libase_201  70818
 . vhd ",
       " ssh_userna  me ":" root ",
       " instance_t  ype ":" ecs . n1 . tiny ",
       " internet_c  harge_type ":" PayByTraff  ic ",
       " io_optimiz  ed ":" true "
     }],
     " provisione  rs ": [{
       " type ": " shell ",
       " inline ": [
         " sleep  30 ",
         " yum  install  redis . x86_64  - y "
       ]
     }]
   }
```

Note:

Customize the values of the following parameters according to your actual requirements.

Parameter	Description

access_key	Your AccessKey ID. For more details, see *create an Accesskey*.
secret_key	Your AccessKey Secret. For more information, see *create an AccessKey*.
region	The region of the temporary instance used to create the custom image.
image_name	The custom image's name
source_image	The name of the basic image name retrieved from Alibaba Cloud public image list.
instance_type	Type of temporary instance generated to create the custom image.
internet_charge_type	The Internet bandwidth billing method for the temporary instance generated for creating the custom image.
provisioners	Type of *Packer Provisioner* used for creating the custom image

Step 3. Create a custom image by using Packer

To specify the Packer template file and create a custom image, follow these steps:

1. Run `export ALICLOUD_A CCESS_KEY = your AccessKeyI D` to import your AccessKey ID.
2. Run `export ALICLOUD_S ECRET_KEY = your AccessKeyS ecret` to import your AccessKey Secret.
3. Run `packer build alicloud . json` to create the custom image.

The following example creates a custom image containing ApsaraDB for Redis and runs as follows:

```
 alicloud - ecs output will be in this color .
==> alicloud - ecs : Prevalidat ing alicloud image name ...
 alicloud - ecs : Found image ID : centos_7_0 2_64_20G_a
 libase_201 70818 . vhd
==> alicloud - ecs : Start creating temporary keypair :
 packer_59e 44f40 - c8d6 - 0ee3 - 7fd8 - b1ba08ea94 b8
==> alicloud - ecs : Start creating alicloud vpc
---------------------------
==> alicloud - ecs : Provisioni ng with shell script : / var
 / folders / 3q / w38xx_js6c l6k5mwkrqs nw7w0000gn / T / packer -
 shell25746 6182
 alicloud - ecs : Loaded plugins : fastestmir ror
---------------------------
```

```
 alicloud - ecs :  Total
   1 . 3   MB / s  |  650   kB   00 : 00
 alicloud - ecs :  Running   transactio  n   check
---------------------------
==>  alicloud - ecs :  Deleting   temporary   keypair ...
 Build  ' alicloud - ecs '  finished .
==>  Builds   finished .  The   artifacts   of   successful   builds
 are :
-->  alicloud - ecs :  Alicloud   images   were   created :
 cn - beijing :  m - 2ze12578be  1oa4ovs6r9
```

What to do next

Use this custom image to create an ECS instance. For more information, see *create an instance from a custom image*.

References

- For more information, visit *packer-provider* , the Packer repository of Alibaba Cloud Github.
- See the *Packer official documents* to learn more about how to use Packer.

3.1.4 Create and import on-premises images by using Packer

Packer is a convenient open-source tool to create on-premises image files. It runs on the most major operating systems.

By using Packer, you can create identical on-premises images for multiple platforms from a single source configuration. This topic details steps to create an on-premises image for CentOS 6.9 on an Ubuntu 16.04 server and to upload it to Alibaba Cloud. For actual scenarios, you can customize your Packer templates as required.

Prerequisites

- You must have an *AccessKey* for the configuration file.

  Note:

  Do not use the AccessKey of your Alibaba Cloud account. Instead, *create a RAM user* and use the RAM account to create the necessary *AccessKey* to maintain account security.

- You must *have purchased OSS*.

Example of creating and importing an on-premises image

1. Run `egrep "( svm | vmx )" / proc / cpuinfo` to check whether your on-premises server or virtual machine supports KVM. If the following output returns, KVM is supported.

```
pat pse36 clflush dts acpi mmx fxsr sse sse2
  ss ht tm pbe syscall nx pdpe1gb rdtscp lm
constant_t sc art arch_perfm on pebs bts rep_good
nopl xtopology nonstop_ts c aperfmperf tsc_known_ freq
  pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2
ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2
x2apic movbe popcnt tsc_deadli ne_timer aes xsave avx
  f16c rdrand lahf_lm abm 3dnowprefe tch epb intel_pt
  tpr_shadow vnmi flexpriori ty ept vpid fsgsbase
tsc_adjust bmi1 avx2 smep bmi2 erms invpcid mpx
rdseed adx smap clflushopt xsaveopt xsavec xgetbv1
  xsaves dtherm ida arat pln pts hwp hwp_notify
hwp_act_wi ndow hwp_epp
flags : fpu vme de pse tsc msr pae mce cx8
  apic sep mtrr pge mca cmov
```

2. Run the following commands to install the KVM:

```
sudo apt - get install qemu - kvm qemu virt - manager
virt - viewer libvirt - bin bridge - utils # Install KVM
and related dependenci es .
sudo virt - manager # Enable virt - manager .
```

If a GUI runs in the VM console window, you have successfully installed the KVM.

3. Install Packer.

To install Packer, see *use Packer to create a custom image*.

4. Run the following commands to define a Packer template.

Note:

The on-premises image created in the following configuration is for the CentOS 6.9 operating system only. To create images for other operating systems, *customize* the configuration file centos.json as required.

```
cd / user / local # Switch the directory .
wget https :// raw . githubuser content . com / alibaba / packer
- provider / master / examples / alicloud / local / centos . json
 # Download file centos . json that is released by
Alibaba Cloud .
wget https :// raw . githubuser content . com / alibaba / packer
- provider / master / examples / alicloud / local / http / centos
- 6 . 9 / ks . cfg # Download file ks . cfg that is
released by Alibaba Cloud .
mkdir - p http / centos - 6 . 9 # Create a directory .
```

```
mv ks . cfg http / centos - 6 . 9 / # Move file ks . cfg
to the http / centos - 6 . 9 directory .
```

5. Run the following commands to create an on-premises image.

```
export ALICLOUD_A CCESS_KEY = SpecifyYou rAccessKey IDHere
# Import your AccessKeyI D ,
export ALICLOUD_S ECRET_KEY = SpecifyYou rAccessKey
SecretHere # Import your AccessKeyS ecret .
packer build centos . json # Create an on - premises
image .
```

An example result is as follows.

```
qemu output will be in this color .
==> qemu : Downloadin g or copying ISO
    qemu : Downloadin g or copying : http :// mirrors .
aliyun . com / centos / 6 . 9 / isos / x86_64 / CentOS - 6 . 9 -
x86_64 - minimal . iso
.....................................................
==> qemu : Running post - processor : alicloud - import
    qemu ( alicloud - import ): Deleting import source
  https :// oss - cn - beijing . aliyuncs . com / packer /
centos_x86 _64
 Build ' qemu ' finished .
==> Builds finished . The artifacts of successful
builds are :
--> qemu : Alicloud images were created :
 cn - beijing : XXXXXXXX
```

6. Wait for a few minutes, log on to the *ECS console* and check your custom image in the image list that is in the corresponding region. In this sample, the region is China North 2 (cn-beijing).

Customize a Packer template

In this example, the following JSON file is customized based on the template used to create an image for the CentOS 6.9.

```
{" variables ": {
        " box_basena me ": " centos - 6 . 9 ",
        " build_time stamp ": "{{ isotime \" 2006010215 0405 \"}}",
        " cpus ": " 1 ",
        " disk_size ": " 4096 ",
        " git_revisi on ": " __unknown_ git_revisi on__ ",
        " headless ": "",
        " http_proxy ": "{{ env ` http_proxy `}}",
        " https_prox y ": "{{ env ` https_prox y `}}",
        " iso_checks um_type ": " md5 ",
        " iso_checks um ": " af4a1640c0 c6f348c6c4 1f1ea9e192 a2
 ",
        " iso_name ": " CentOS - 6 . 9 - x86_64 - minimal . iso ",
        " ks_path ": " centos - 6 . 9 / ks . cfg ",
        " memory ": " 512 ",
        " metadata ": " floppy / dummy_meta data . json ",
        " mirror ": " http :// mirrors . aliyun . com / centos ",
        " mirror_dir ectory ": " 6 . 9 / isos / x86_64 ",
        " name ": " centos - 6 . 9 ",
```

```
        " no_proxy ": "{{ env  ` no_proxy `}}",
        " template ": " centos - 6 . 9 - x86_64 ",
        " version ": " 2 . 1 . TIMESTAMP "
      },
    " builders ":[
      {
        " boot_comma  nd ": [
          "< tab >  text   ks = http ://{{ . HTTPIP  }}:{{ .
 HTTPPort  }}/{{ user  ` ks_path `}}< enter >< wait >"
        ],
        " boot_wait ": " 10s ",
        " disk_size ": "{{ user  ` disk_size `}}",
        " headless ": "{{  user  ` headless ` }}",
        " http_direc  tory ": " http ",
        " iso_checks  um ": "{{ user  ` iso_checks  um `}}",
        " iso_checks  um_type ": "{{ user  ` iso_checks  um_type
 `}}",
        " iso_url ": "{{ user  ` mirror `}}/{{ user  ` mirror_dir
 ectory `}}/{{ user  ` iso_name `}}",
        " output_dir  ectory ": " packer -{{ user  ` template `}}-
 qemu ",
        " shutdown_c  ommand ": " echo  ' vagrant '| sudo  - S  /
 sbin / halt  - h  - p ",
        " ssh_passwo  rd ": " vagrant ",
        " ssh_port ":  22 ,
        " ssh_userna  me ": " root ",
        " ssh_wait_t  imeout ": " 10000s ",
        " type ": " qemu ",
        " vm_name ": "{{  user  ` template ` }}. raw ",
        " net_device ": " virtio - net ",
        " disk_inter  face ": " virtio ",
        " format ": " raw "
      }
    ],
    " provisione  rs ": [{
        " type ": " shell ",
        " inline ": [
            " sleep   30 ",
            " yum   install   cloud - util   cloud - init  - y "
          ]
      }],
    " post - processors ":[
      {
        " type ":" alicloud - import ",
        " oss_bucket  _name ": " packer ",
        " image_name ": " packer_imp  ort ",
        " image_os_t  ype ": " linux ",
        " image_plat  form ": " CentOS ",
        " image_arch  itecture ": " x86_64 ",
        " image_syst  em_size ": " 40 ",
        " region ":" cn - beijing "
      }
    ]
}
```

Parameters in a Packer builder

QEMU builder is used in the preceding *example* to create a virtual machine image. Required parameters for the builder are as follows.

Parameter	Type	Description
iso_checksum	String	The checksum for the OS ISO file. Packer verifies this parameter before starting a virtual machine with the ISO attached. Make sure you specify at least one of the `iso_checksum` or `iso_checksum_url` parameters. If you have specified the `iso_checksum` parameter, the `iso_checksum_url` parameter is automatically ignored.
iso_checksum_type	String	The type of the checksum specified in iso_checksum. Optional values: · none: If you specify none for iso_checksum_type, the checksuming is ignored. This value is not recommended. · md5 · sha1 · sha256 · sha512
iso_checksum_url	String	A URL that points to a GNU or BSD style checksum file that contains the ISO file checksum of an operating system. It may come in either the GNU or BSD pattern. Make sure you specify either the `iso_checksum` or the `iso_checksum_url` parameter. If you specify the iso_checksum parameter, the `iso_checksum_url` parameter is automatically ignored.
iso_url	String	A URL that points to the ISO file, and contains the installation image. This URL may be an HTTP URL or a file path: · If it is an HTTP URL, Packer downloads the file from the HTTP link and caches the file for later . · If it is a file path to the IMG or QCOW2 file, QEMU directly starts the file. If you have the file path specified, set parameter `disk_image` to `true` .
headless	boolean	By default, Packer starts the virtual machine GUI to build a QEMU virtual machine. If you set `headless` to `True` , a virtual machine without any console is started.

For more information, see Packer *QEMU Builder*.

Parameters in a Packer provisioner

The provisioner in the preceding *example* contains a Post-Processor module that enables automated upload of on-premises images to Alibaba Cloud. Required parameters for the provisioner are as follows:

Parameter	Type	Description
access_key	String	Your AccessKeyID. The AccessKey has a high privilege. We recommend that you first *create a RAM user* and use the RAM account to create an AccessKey to maintain security of your Alibaba Cloud account.
secret_key	String	Your AccessKeySecret. The AccessKey has a high privilege. We recommend that you first *create a RAM user* and use the RAM account to create an AccessKey to maintain security of your Alibaba Cloud account.
region	String	Select the region where you want to upload your on-premises image. In this example, the region is cn-beijing. For more information, see *regions and zones*.
image_name	String	The name of your on-premises image. The name is a string of 2 to 128 characters. It must begin with an English or a Chinese character. It can contain A-Z, a-z, Chinese characters, numbers , periods (.), colons (:), underscores (_), and hyphens (-).
oss_bucket _name	String	The OSS bucket name. If you specify a bucket name that does not exist, Packer creates a bucket automatically with the specified oss bucket name when uploading the image.
image_os_type	String	Image type. Optional values: · linux · windows
image_plat form	String	Distribution of the image. For example, CentOS.

Parameter	Type	Description
image_architecture	String	The instruction set architecture of the image. Optional values: · i386 · x86_64
format	String	Image format. Optional values: · RAW · VHD

For more information, see Packer *Alicloud Post-Processor*.

Next step

You can use the created image to create an ECS instance. For more information, see *create an instance from a custom image*.

References

- For more information about how to use Packer, see *Packer* documentation.
- For more information about release information, visit the Packer repository on GitHub *packer*.
- For more information about Alibaba Cloud open source tools, visit the Alibaba Cloud repository on GitHub *opstools*.
- For more information about Alibaba Cloud and Packer project, visit the Alibaba Cloud & Packer repositories on GitHub *packer-provider*.
- For more information about configuration file ks.cfg, see *Anaconda Kickstart* .

3.2 Manage custom images

You can modify the name and description of your custom images to help you organize and identify them, and you can delete custom images that you no longer require

Modify the name and description of a custom image

To modify the name and description of a custom image, follow these steps:

1. Log on to the *ECS console*.
2. In the left-side navigation pane, select Snapshots and Images > Images.
3. Select the target region.

4. Find the custom image to be edited and then click the icon..
5. Enter a name for the custom image.

6. In the Actions column, click Modify Description and then, in the dialog box, enter a Custom Image Description.
7. Click Save.

Alternatively, you can modify the name and description of a custom image by calling the ECS API `ModifyImageAttribute`.

Delete custom images

To delete one or more custom images, follow these steps:

1. Log on to the *ECS console*.
2. In the left-side navigation pane, select Snapshots and Images > Images.
3. Select the target region.
4. Select one or more custom images that you want to delete, and then click Delete.

5. In the dialog box that appears, select the required method for deleting the custom images:
   - Delete: The custom images are deleted normally.
   - Force Delete: The custom images are deleted forcibly. Check I confirm to forcibly Delete the selected instances.

     Note:
     After you forcibly delete the custom images, *cloud disk reinitialization* of the instances that you have created from the images cannot be performed.

6. Click OK.

Alternatively, you can delete custom images by calling the ECS API `DeleteImage`.

3.3 Import images

3.3.1 Image compliance tool

This topic introduces how to use the image compliance tool to automatically locate the operating system settings of non-Alibaba Cloud specification through operation examples, parameter description, and output details. The tool is suitable for importing custom images scenarios.

Introduction

ECS allows you to create instances from imported custom images. Imported custom images can be created based on your offline server, virtual machine, or a cloud host on any cloud platform. The images you import must meet certain requirements. For more information, see *Notes for importing images*.

To reduce the time required for creating images and instances, we recommend that you use the image compliance tool of ECS (referenced in this document as compliance tool) to create images that comply with the relevant standards. The compliance tool can detect non-compliance of various configuration indicators and locations based on a given server environment, generate TXT and JSON detection reports, and offer possible solutions.

Limits

The compliance tool currently supports Linux images only, such as Ubuntu, CentOS, Debian, RedHat, SUSE Linux Enterprise Server (SLES), OpenSUSE, FreeBSD, CoreOS, and other Linux versions.

Sample

The following example use a CentOS 7.4 64-bit server.

1. Log on to your server, virtual machine, or cloud host.
2. *Download* the compliance tool.
3. Run `image_chec k` with root permissions to guarantee that the compliance tool can read configuration files under permission control.

```
chmod + x image_chec k
sudo image_chec k - p [ destinatio n path ]
```

 Note:

You can use `-p [destination path]` to specify the path where detection reports are generated. If you do not set this parameter, reports are generated in the compliance tool path by default.

4. Wait for the compliance tool to detect the system configuration.

```
Begin check your system ...
 The report is generating .
-----------------------------------------
 The information you need to enter when you
import your image to the Alibaba Cloud website :
 Current system : CentOS # System information 1 :
Server operating system
 Architecture : x86_64 # System information 2 : System
  architecture
 System disk size : 42 GB # System information 3 :
Server system disk capacity
-----------------------------------------
# Detection item
 Check driver [ OK ]
 Check shadow file authority [ OK ]
 Check security [ OK ]
 Check qemu-ga [ OK ]
 Check network [ OK ]
 Check ssh [ OK ]
 Check firewall [ OK ]
 Check filesystem [ OK ]
 Check device id [ OK ]
 Check root account [ OK ]
 Check password [ OK ]
 Check partition table [ OK ]
 Check lvm [ FAILED ]
 Check lib [ OK ]
 Check disk size [ OK ]
 Check disk use rate [ WARNING ]
 Check inode use rate [ OK ]
-----------------------------------------
 15 items are OK
 1 items are failed
 1 items are warning
-----------------------------------------
 The report is generated : / root / image_check_report_2
018 - 05 - 14_18 - 18 - 10 . txt
 Please read the report to check the details
```

5. View the detection report. The report is generated in the format of `image_check_report_date_time.txt` or `image_check_report.json`.

Detection items

The compliance tool detects the following server configuration items to ensure that the ECS instances created from your custom image are fully functional.

Detection item	Non-compliance issue	Suggestion
driver	The ECS instance cannot start normally.	Install a virtualization driver. For example, *install a virtio driver*

Detection item	Non-compliance issue	Suggestion
/etc/shadow	You cannot modify the password file, so you cannot create an ECS instance from the custom image.	Do not use the `chattr` command to lock the /etc/shadow file.
SElinux	The ECS instance cannot start normally.	Do not modify /etc/selinux/config to start SELinux.
qemu-ga	Some of the services required by ECS are unavailable, and the instance is not fully functional.	Uninstall qemu-ga.
network	Network functions of the ECS instance are unstable.	Disable or delete the Network Manager and enable the network service.
ssh	You cannot *connect* to the ECS instance from the console.	Enable the SSH service and do not set PermitRootLogin.
firewall	The system does not automatically configure your ECS instance environment.	Disable the firewall iptables, firewalld, IPFilter (IPF), IPFireWall (IPFW), or PacketFilter (PF).
file system	You cannot *resize the disk*.	The XFS, Ext3, and Ext4 file systems are used, and the Ext2, UFS, and UDF file systems are allowed. The Ext4 file system does not support 64-bit features.
root	You cannot use your username and password to remotely connect to the ECS instance.	Reserve the root account.
passwd	You cannot add users to the ECS instance.	Retain or reinstall the passwd command.
Partition table	The ECS instance cannot start normally.	Use MBR partitioning.
Logical Volume Manager (LVM)	The ECS instance cannot start normally.	Switch to another partitioning service.
/lib	The ECS instance cannot be automatically configured.	The /lib and /lib64 files cannot be stored in absolute paths. Modify the storage paths of /lib and /lib64 to their relative paths.

Detection item	Non-compliance issue	Suggestion
system disk	N/A	Increase the system disk capacity. The optimal system disk capacity is 40 GiB to 500 GiB. When you import images, configure the system disk capacity based on the virtual file size of images, instead of the usage capacity of images.
disk_usage	You cannot install the necessary drivers or services for the ECS instance.	Make sure that sufficient disk space is available.
inode usage	You cannot install the necessary drivers or services for the ECS instance.	Make sure that sufficient inode resources are available.

The compliance tool provides a detection result `OK`, `FAILED`, or `WARNING` based on detection items.

- `OK`: The detection items all comply with requirements.
- `FAILED`: The detection items do not comply with requirements, which means a ECS instance created from the custom image cannot start normally. We recommend that you rectify the non-compliant items and recreate the image to improve instance startup efficiency.
- `WARNING`: The detection items do not comply with requirements, which means an ECS instance created from the custom image can start normally, but ECS cannot use valid methods to configure your instance. You can choose to immediately rectify the non-compliant items or temporarily retain the items and create an image.

Output items

The compliance tool provides detection reports in both TXT and JSON formats after it detects the system environment. You can use `-p [destination path]` to specify the path where detection reports are generated. If you do not specify this parameter, reports are generated in the compliance tool path by default.

- Reports in TXT format are named `image_chec k_report_d ate_time . txt`. The reports include server configuration information and detection results. The following example uses a CentOS 7.4 64-bit server.

```
The informatio n you need to input when you import
  your image to Alibaba Cloud Website :
  Current system is : CentOS # Server operating system
  Architectu re : x86_64 # System architectu re
  System disk size : 42 GB # Server system disk
capacity
 ----------------------------------------
   Check driver # Detection item name
   Pass : kvm drive is exist # Detection result
   Alibaba Cloud supports kvm virtualiza tion
technology
   We strongly recommend installing kvm driver .
```

- Reports in JSON format are named `image_chec k_report . json`. The reports include server configuration information and detection results. The following example uses a CentOS 7.4 64-bit server.

```
" platform ": " CentOS ", \\ Server operating system
 " os_big_ver sion ": " 7 ", \\ Operating system version
number ( major )
 " os_small_v ersion ": " 4 ", \\ Operating system version
number ( minor )
 " architectu re ": " x86_64 ", \\ System architectu re
 " system_dis k_size ": " 42 ", \\ Server system disk
capacity
 " version ": " 1 . 0 . 2 ", \\ Compliance tool version
 " time ": " 2018 - 05 - 14_19 - 18 - 10 ", \\ Detection time
 " check_item s ": [{
     " name ": " driver ", \\ Detection item name
     " result ": " OK ", \\ Detection result
     " error_code ": " 0 ", \\ Error code
     " descriptio n ": " Pass : kvm driver exists .", \\
Descriptio n
     " comment ": " Alibaba Cloud supports kvm virtualiza
tion technology . We strongly recommend installing kvm
driver ."
  }]
}
```

What to do next

1. View the *notes for importing images*.
2. *Install the virtio driver.*
3. (Optional) *Convert the image file format.*
4. *Import custom images.*
5. *Create an instance from a custom image.*

3.3.2 Notes for importing images

To guarantee the usability of an imported image and improve the importing efficiency, the following considerations must be noted before importing an image:

Depending on the operating system, the notes vary for *Windows images* and *Linux images*.

Windows images

Considerations

- Verify the integrity of the file system before importing images for Windows.
- Check that there is adequate space on the system disk for the image to be installed.
- Disable the firewall and allow access to RDP port 3389.
- The logon password for the administrator account must be 8-30 characters in length and can contain letters, numbers, and the following special characters ( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ‘ < > , . ? /
- Configure the system disk size for the importing based on the virtual disk size rather than the usage of the image. The size of the disk to be used for the image import must be a minimum of 40 GiB, and cannot exceed 500 GiB.
- Do not modify critical system files.

What are supported

- Multi-partition system disks.
- NTFS file systems and MBR partitions.
- Images in RAW, qcow2, or VHD format.

Note:

If you want to import an image in another format, you need to *convert image file format* before importing it. We recommended that you convert the format to VHD because it offers smaller transmission capacity.

- Images with the following operating system versions can be imported:
  - Microsoft Windows Server 2016
  - Microsoft Windows Server 2012 R2 (standard edition)
  - Microsoft Windows Server 2012 (standard edition and data center edition)
  - Microsoft Windows Server 2008 R2 (standard edition, data center edition, and enterprise edition)
  - Microsoft Windows Server 2008 (standard edition, data center edition, and enterprise edition)
  - Microsoft Windows Server 2003 with Service Pack 1 (SP1) (standard edition, data center edition, and enterprise edition) or higher

What are not supported

- The installation of qemu-ga in an image is not supported because some services needed by ECS will become unavailable.
- Windows XP, Windows 7 (professional and enterprise editions), Windows 8, and Windows 10.

Linux images

Considerations

- Verify the integrity of the file system before importing images for Linux.
- Check that there is adequate space on the system disk for the image to be installed.
- Disable the firewall and allow access to TCP port 22.
- Install the virtualization platform XEN or KVM drives.
- We recommended that you *install cloud-init*, so as to guarantee that hostname, NTP, and yum sources can be configured successfully.
- Dynamic Host Configuration Protocol (DHCP) needs to be enabled.
- The logon password for the root account must be 8-30 characters long and must contain uppercase/lowercase letters, numbers, and special characters simultaneo usly. The special characters can be: ( ) ` ~ ! @ # $ % ^ & * - + = | { } [ ] : ; ‘ < > , . ? /
- Do not modify critical system files, such as `/ sbin` , `/ bin` , and `/ lib *`.

What aresupported

- Images in RAW, qcow2, or VHD format.

Note:

If you want to import an image in another format, you need to *convert image file format* before importing it. We recommended that you convert the format to VHD because it offers smaller transmission capacity.

- The xfs, ext3, and ext4 file systems and MBR partitions.

 Note:
The ext4 file system cannot include the `64bit` feature. Moreover, the `project` and `quota` features cannot appear in pairs. You can run the command `tune2fs -l < ext4 file system directory > | grep features` to view the features included in the ext4 file system.

What are not supported

- Multiple network interfaces.
- IPv6 addresses.
- System disk partitions cannot be adjusted. Currently, only a single root partition is supported.

Non-standard image usage notes

Depending on whether the Linux system image you are importing is a standard platform image, the following issues must be noted.

- Official operating system releases are defined as *standard platform images*. Currently, supported system releases include Aliyun Linux, CentOS 5/6/7, CoreOS 681.2.0+, Debian 6/7, FreeBSD, OpenSUSE 13.1, RedHat, Red Hat Enterprise Linux (RHEL), SUSE Linux 10/11/12, and Ubuntu 10/12/13/14.
- Operating system images that are not listed as public images provided by ECS are *non-standard platform images*. Such images, though based on the standard operating system, do not comply with the requirements for a standard operating system regarding critical system configuration files, basic system environments, and applications. If you want to use a non-standard platform image, you can only choose the following when importing an image:
  - Other Linux: Alibaba Cloud identifies all of these images as other Linux systems. Alibaba Cloud does not handle the instances created if you import an image of Other Linux type. If you enable DHCP before creating an image, Alibaba Cloud automatically configures your network. After creating the instance, you need to

connect to the instance by using the *Management Terminal* feature in the console, and then manually configure the IP address, router, and password.

- Customized Linux: Customized images. After importing a customized Linux image, configure the network and password of the instance according to the standard system configuration mode of Alibaba Cloud. For more information, see *customize Linux images*.

Item	Standard platform image	Non-standard platform image
Requirements for critical system configuration files	· Do not modify `/ etc / issue *`. Otherwise, ECS cannot properly identify the system release, leading to system creation failure. · Do not modify `/ boot / grub / menu . lst` , or the ECS instance cannot be started. · Do not modify `/ etc / fstab` , or the exception partition cannot be loaded, leading to ECS instance start failure. · Do not change `/ etc / shadow` to read only, or you may be unable to modify the password file, leading to system creation failure. · Do not enable SELinux by modifying `/ etc / selinux / config` , or the system may fail to start.	Does not meet the requirements of standard platform images
Requirements for applications	Do not install qemu-ga in an imported image, or some services required by Alibaba Cloud may become unavailable.	Does not meet the requirements of standard platform images

3.3.3 Install cloud-init for Linux images

When you use a custom Linux image, we recommend that you install cloud-init in your servers to guarantee successful initialization of the instances running that image.

What is cloud-init?

cloud-init is an open source software used by cloud-based platforms to configure system initialization of Linux instances. It is supported by major platforms such

as Alibaba Cloud, AWS, Azure, and OpenStack. For more information, see *cloud-init documentation*.

Alibaba Cloud cloud-init initializes the configurations of instances during their startup, including the network, NTP, software source, host name, and SSH key pair. It also executes the *user data* script.

Scenarios

Cloud-init is installed by default for all public images of Alibaba Cloud. If you use custom images in the following scenarios, we recommend that you install Alibaba Cloud cloud-init for your Linux servers so as to ensure the system configurations of created instances can be automatically initialized:

- Linux servers that will be migrated to Alibaba Cloud, but have not installed cloud-init.
- Linux servers that have installed cloud-init whose version is earlier than 0.7.9.
- Alibaba Cloud ECS instances that have not installed cloud-init.

Check the cloud-init version

Different cloud platforms may use different versions of cloud-init. Please select the appropriate version and configure the appropriate datasource. Alibaba Cloud uses cloud-init 0.7.6a and the data source is `Aliyun`.

After cloud-init is installed, its option of self-start upon instance startup is enabled by default. If the selected cloud-init version or data source is not suitable, cloud-init may run abnormally and the instance may start slowly (or fail to start) the next time you restart your instance. As a result, we recommend that you back up your data before you install it. Exercise caution when you install it if you are not fully ready to migrate your servers onto Alibaba Cloud.

- Check if cloud-init is installed: `which cloud - init`

   Note:
  No output indicates that it is not installed and you need to install the Alibaba Cloud cloud-init.

- Check the cloud-init version: `cloud - init -- version`

   Note:

If the version is earlier than the community version 0.7.9 (except 0.7.6a), you need to install the Alibaba Cloud cloud-init.

(Recommended) Install the Alibaba Cloud cloud-init

1. Check and install the python-pip dependency.
2. *Download the Alibaba Cloud cloud-init* and decompress it to the current directory:

```
wget http :// ecs - image - utils . oss - cn - hangzhou .
aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
tar - zxvf ali - cloud - init - latest . tgz
```

3. Enter the `tools` directory of `cloud - init` , and then run the cloud-init installation script `deploy . sh` :

```
bash ./ deploy . sh < issue > < major_vers ion >
```

- The parameters are described as follows:
  - `issue` : The operating system platform. The value range is: `centos | redhat | rhel | debian | ubuntu | opensuse | sles` . The parameter values are case sensitive. `sles` represents SUSE/SLES.
  - `major_vers ion` : The major version of an operating system platform. For example, the major version of CentOS 6.5 is `6` .
- The following are command examples:
  - Install cloud-init in CentOS 6.5:

```
bash ./ deploy . sh centos 6
```

  - Install cloud-init in Ubuntu 14.04:

```
bash ./ deploy . sh ubuntu 14
```

4. Confirm that the installation is successful. If `" descriptio n ": " success "` is returned, the installation is successful.

Install the Alibaba Cloud cloud-init in different platforms

Installation commands for different platforms are shown as follows:

- CentOS 6/7

```
# Check and install python - pip
if ! python - c ' import setuptools ' >& / dev / null ;
then
  yum - y install python - pip
```

```
 fi
#  Back   up   the   legacy   cloud - init   configurat  ion
 test  - d  / etc / cloud  &&  mv  / etc / cloud  / etc / cloud -
 old
#  Download   and   decompress   the   Alibaba   Cloud   cloud -
 init
 wget   http :// ecs - image - utils . oss - cn - hangzhou .
 aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
 tar  - zxvf  ./ ali - cloud - init - latest . tgz
#  Install   cloud - init
 issue_majo  r =$(  cat  / etc / redhat - release  |  awk  '{ printf
  $ 3 }' |  awk  - F '.' '{ printf  $ 1 }')
 bash  ./ cloud - init -*/ tools / deploy . sh   centos  "$
 issue_majo  r "
```

- **RHEL 6/7**

```
#  Check   and   install   python - pip
 if  !  python  - c  ' import   setuptools ' >& / dev / null ;
 then
   yum  - y   install   python - pip
 fi
#  Back   up   the   legacy   cloud - init   configurat  ion
 test  - d  / etc / cloud  &&  mv  / etc / cloud  / etc / cloud -
 old
#  Download   and   decompress   the   Alibaba   Cloud   cloud -
 init
 wget   http :// ecs - image - utils . oss - cn - hangzhou .
 aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
 tar  - zxvf  ./ ali - cloud - init - latest . tgz
#  Install   cloud - init
 issue_majo  r =$( cat  / etc / os - release  |  grepVERSIO  N_ID
  |  awk  - F '"' '{ printf  $ 2 }' |  awk  - F '.' '{ printf  $ 1
 }')
 bash  ./ cloud - init -*/ tools / deploy . sh   rhel  "$
 issue_majo  r "
```

- **Ubuntu 14/16/18**

```
#  Check   and   install   python - pip
 if  !  python  - c  ' import   setuptools ' >& / dev / null ;
 then
   apt - get   install   python - pip  - y
 fi
#  Back   up   the   legacy   cloud - init   configurat  ion
 test  - d  / etc / cloud  &&  mv  / etc / cloud  / etc / cloud -
 old
#  Download   and   decompress   the   Alibaba   Cloud   cloud -
 init
 wget   http :// ecs - image - utils . oss - cn - hangzhou .
 aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
 tar  - zxvf  ./ ali - cloud - init - latest . tgz
#  Install   cloud - init
 issue_majo  r =$( cat  / etc / os - release  |  grep   VERSION_ID
  |  awk  - F '"' '{ printf  $ 2 }' |  awk  - F '.' '{ printf  $ 1
 }')
 bash  ./ cloud - init -*/ tools / deploy . sh   ubuntu  "$
 issue_majo  r "
```

- **Debian 8/9**

```
#  Check   and   install   python - pip
```

```
if ! python -c 'import setuptools' >&/dev/null;
then
  apt-get -y install python-pip
fi
# Back up the legacy cloud-init configurat ion
test -d /etc/cloud && mv /etc/cloud /etc/cloud-
old
# Download and decompress the Alibaba Cloud cloud-
init
wget http://ecs-image-utils.oss-cn-hangzhou.
aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# Install cloud-init
issue_majo r=$(cat /etc/os-release | grep VERSION_ID
 | awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
bash ./cloud-init-*/tools/deploy.sh debian "$
issue_majo r"
```

- SUSE 11/12

```
# Check and install python-pip
if ! python -c 'import setuptools'>&/dev/null; then
  zypper -n install python-pip
fi
# Back up the legacy cloud-init configurat ion
test -d /etc/cloud && mv /etc/cloud /etc/cloud-
old
# Download and decompress the Alibaba Cloud cloud-
init
wget http://ecs-image-utils.oss-cn-hangzhou.
aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# Install cloud-init
issue_majo r=$(cat /etc/os-release | grepVERSIO N_ID
 | awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
bash ./cloud-init-*/tools/deploy.sh sles "$
issue_majo r"
```

- OpenSUSE 13/42

```
# Check and install python-pip
if ! python -c 'import setuptools'>&/dev/null; then
  zypper -n install python-pip
fi
# Back up the legacy cloud-init configurat ion
test -d /etc/cloud && mv /etc/cloud /etc/cloud-
old
# Download and decompress the Alibaba Cloud cloud-
init
wget http://ecs-image-utils.oss-cn-hangzhou.
aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# Install cloud-init
issue_majo r=$(cat /etc/os-release | grepVERSIO N_ID
 | awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
```

```
bash ./ cloud - init -*/ tools / deploy . sh opensuse "$
issue_majo r "
```

(Optional) Install the community cloud-init

Prerequisites

Before installing cloud-init, you must install the git, python 2.7, and python-pip dependencies. The installation commands are as follows:

- **CentOS/RedHat:**

```
yum - y install git python python - pip
```

- **Ubuntu/Debian:**

```
apt - get - y install git python python - pip
```

- **OpenSUSE/SUSE:**

```
zypper - n install git python python - pip
```

Procedure

1. **Log on to the source server.**
2. **Download the cloud-init package through git and enter the cloud-init directory:**

```
git clone https :// git . launchpad . net / cloud - init
cd ./ cloud - init
```

3. **Install all the dependencies:**

```
pip install - r ./ requiremen ts . txt
```

4. **Install cloud-init:**

```
python setup . py install
```

5. **Run `vi / etc / cloud / cloud . cfg` to modify the configuration file cloud.cfg.**

   Modify the configurations of `cloud_init _modules :` as follows:

```
# Example datasource config
# The top level settings are used as module
# and system configurat ion .
# A set of users which may be applied and / or
 used by various modules
# when a ' default ' entry is found it will reference
 the ' default_us er '
# from the distro configurat ion specified below
 users :
   - default
```

```
 user :
     name :  root
     lock_passw  d :  False
#  If   this   is   set , ' root '  will   not   be   able   to
 ssh   in   and   they
#  will   get   a   message   to   login   instead   as   the
 above  $ user
 disable_ro  ot :  false
#  This   will   cause   the   set + update   hostname   module   to
   not   operate  ( if   true )
 preserve_h  ostname :  false
 syslog_fix  _perms :  root : root
 datasource  _list : [  AliYun  ]
#  Example   datasource   config
 datasource :
     AliYun :
         support_xe  n :  false
         timeout :  5  # ( defaults   to   50   seconds )
         max_wait :  60  # ( defaults   to   120   seconds )
#       metadata_u  rls : [ ' blah . com ' ]
#  The   modules   that   run   in   the   ' init '  stage
 cloud_init  _modules :
```

What to do next

- For Linux servers that are ready for migration to the cloud, you can *migrate them to Alibaba Cloud by using the Cloud Migration tool* or *import custom images*.
- For ECS instances with custom Linux images that are already running on Alibaba Cloud, you can restart the system for verification. If the system is automatically configured with the host name, software source, and NTP, cloud-init is successfully installed.

Troubleshooting

Note:

- The libraries that are missing may vary according to the images. To resolve this issue, you can install them through pip, and then install cloud-init again.
- If the default package manager (for example `yum`) and the `pip` manager have installed different versions of libraries, library version conflicts may occur and lead to cloud-init running abnormally. We recommend that you download the dependency libraries according to the reported errors.

Error: no setuptools module in python

The error prompt `no setuptools module in python` means you need to install python setuptools. You can fix it in the following ways:

- CentOS/RedHat: `yum -y install python-pip`

- **Ubuntu/Debian:** `apt - get - y install python - pip`
- **OpenSUSE/SUSE:** `zypper - n install python - pip`

Error: No module named six

Run `pip install six` **to resolve the following error.**

```
File "/ root / cloud - init / cloudinit / log . py ", line 19 ,
in < module >
    import six
ImportErro r : No module named six )
```

Error: No module named oauthlib.oauth1

Run `pip install oauthlib` **to resolve the following error.**

```
File "/ root / cloud - init / cloudinit / url_helper . py ", line
20 , in < module >
    import oauthlib . oauth1 as oauth1
ImportErro r : No module named oauthlib . oauth1 )
```

Error messages do not indicate the missing libraries

If error messages do not indicate the missing libraries during installation, you can install all the dependency libraries displayed in the file requirements.txt of cloud-init.

```
pip install - r requiremen ts . txt
```

References

cloud-init website - Alibaba Cloud (AliYun)

3.3.4 Install virtio driver

This topic details which images do and do not require the virtio driver to be installed on the source server before import.

Images requiring no manual installation

After you *import custom images*, **if the operating systems of your images are listed as follows, Alibaba Cloud automatically processes the virtio driver for you:**

- **Windows Server 2008**
- **Windows Server 2012**
- **Windows Server 2016**
- **CentOS 6/7**
- **Ubuntu 12/14/16**
- **Debian 7/8/9**

- SUSE 11/12

You can skip to recover the temporary root file system of initramfs or initrd.

Images requiring manual installation

For Linux images that are not included in the preceding list, you must install the virtio driver on-premises before importing the images.

To check the availability of virtio driver on a server

1. Run `grep -i virtio /boot/config-$(uname -r)` to inspect whether the virtio driver is already built in the kernel of your server.

Note:

- If VIRTIO_BLK and VIRTIO_NET do not exist in the output, the virtio driver is not built in the kernel. You must install and configure the virtio driver on your server *to compile and install virtio driver*.
- If the values of parameter `CONFIG_VIRTIO_BLK` and parameter `CONFIG_VIRTIO_NET` are `y`, the virtio driver is already built in the kernel. For more information, see *notes for importing images* and *import custom images*.
- If the values of parameter `CONFIG_VIRTIO_BLK` and parameter `CONFIG_VIRTIO_NET` are `m`, continue to step 2.

2. Run `lsinitrd /boot/initramfs-$(uname -r).img | grep virtio` to make sure the virtio driver has been complied in the temporary root file system of initramfs or initrd.

Note:

- According to the preceding figure, the virtio_blk driver, including its dependency virtio.ko, virtio_pci.ko and virtio_ring.ko, has been compiled in the temporary root file system initramfs. For more information, see *notes for importing images* and *import custom images*.
- If virtio driver is unavailable in the initramfs, you must recover the temporary root file system of initramfs or initrd before importing images or migration.

To recover the temporary root file system

If the virtio driver is supported by the kernel but not compiled in the temporary root file system, you must recover the temporary root file system. The following example uses CentOS:

- CentOS/RedHat 5

```
mkinitrd - f -- allow - missing \
          -- with = xen - vbd -- preload = xen - vbd \
          -- with = xen - platform - pci -- preload = xen -
platform - pci \
          -- with = virtio_blk -- preload = virtio_blk \
          -- with = virtio_pci -- preload = virtio_pci \
          -- with = virtio_con sole -- preload = virtio_con
sole \
```

- CentOS/RedHat 6/7

```
mkinitrd - f -- allow - missing \
          -- with = xen - blkfront -- preload = xen - blkfront \
          -- with = virtio_blk -- preload = virtio_blk \
          -- with = virtio_pci -- preload = virtio_pci \
          -- with = virtio_con sole -- preload = virtio_con
sole \
          / boot / initramfs -$( uname - r ). img $( uname - r
)
```

- Debian/Ubuntu

```
echo - e ' xen - blkfront \ nvirtio_bl k \ nvirtio_pc i \
nvirtio_co nsole ' >> \
/ etc / initramfs - tools / modules
mkinitramf s - o / boot / initrd . img -$( uname - r )"
```

To compile and install virtio driver

The following example uses a Red Hat server:

To download the kernel package

1. Run `yum install - y ncurses - devel gcc make wget` to install necessary components to compile the kernel.
2. Run `uname - r` to query the kernel version of your server, such as 4.4.24-2.a17.x86_64.
3. Visit *published Linux Kernel Archives* to download the source codes of kernel, for example, the download link of kernel version starting with 4.4.24 is *https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.4.24.tar.gz*.
4. Run `cd / usr / src /` to change the directory.

5. Run `wget https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.4.24.tar.gz` to download the installation package.
6. Run `tar -xzf linux-4.4.24.tar.gz` to decompress the package.
7. Run `ln -s linux-4.4.24 linux` to establish a link.
8. Run `cd /usr/src/linux` to change the directory.

To compile the kernel

1. Run the following commands to compile the driver into the kernel.

```
make mrproper
symvers_path=$(find /usr/src/ -name "Module.symvers")
test -f $symvers_path th && cp $symvers_path th .
cp /boot/config-$(uname -r) ./.config
make menuconfig
```

2. Configure the corresponding settings of virtio driver in the following windows:

> **Note:**
> Select * to build the driver in the kernel, select `m` to compile it as a module.

a. Press the space bar to select Virtualization.

Make sure that you have selected the option of KVM (Kernel-based Virtual Machine).

```
Processor type and features --->
  [*] Paravirtualized guest support --->
    --- Paravirtualized guest support
(128) Maximum allowed size of a domain in gigabytes
[*]   KVM paravirtualized clock
[*]   KVM Guest support
```

```
Device Drivers --->
 [*] Block devices --->
<M>   Virtio block driver (EXPERIMENTAL)
-*- Network device support --->
```

```
      < M >    Virtio   network   driver  ( EXPERIMENT  AL )
```

b. Press the Esc key to exit the kernel configuration windows, and save changes to file .config according to the dialog box.

c. Inspect whether all the corresponding settings of virtio driver has been correctly configured or not.

d. (Optional) If no configuration of virtio driver is settled after the inspect, run the following commands to edit the file .config manually.

```
make   oldconfig
make   prepare
make   scripts
make
make   install
```

e. Run the following commands to check whether the virtio driver is installed.

```
find  / lib / modules /"$( uname  - r )"/ - name  " virtio .*" |
grep  - E  " virtio .*"
grep  - E  " virtio .*" < / lib / modules /"$( uname  - r )"/
modules . builtin
```

Note:

If any of the output includes virtio_blk and virtio_pci.virtio_console, your server has correctly installed the virtio driver.

What to do next

After compiling the virtio driver, you can *migrate your server to Alibaba Cloud by using the Cloud Migration Tool*.

3.3.5 Customize Linux images

If your selected OS is not supported by Alibaba Cloud, and cloud-int cannot be installed, you can select Customized Linux when importing a customized image. Alibaba Cloud will then regard the customized Linux image as an unrecognized OS type (that is, it lacks necessary standard configuration information for ECS instance start for the first time). In this case, you need to add a parsing script to the customized image before importing the image, so as to facilitate automatic configuration of the instance at the first start.

Limitations

- The first partition of the customized Linux image must be writable.

- The first partition type of the customized Linux image must be FAT32, EXT2, EXT3, EXT4, or UFS.
- The size of the virtual file of the customized Linux image must be larger than 5 GiB.
- Security requirements for customized Linux images are as follows:
  - There is no high-risk vulnerability that can be remotely exploited.
  - When you log on to an instance for the first time through the ***Management Terminal*** of the ECS console, you must change the initial default password (if there is any) before performing subsequent actions.
  - There is no default SSH private key pair. The initial SSH private key pair must be randomly generated by Alibaba Cloud.

Procedure

1. Create the `aliyun_cus tom_image` directory in the root directory of the first image partition.

   When the instance that is created using the customized Linux image is started for the first time, Alibaba Cloud will write configuration information into the os.conf file in the `aliyun_cus tom_image` directory. Alibaba Cloud will automatically create an os.conf file if there is none.
2. Create a parsing script in the image to parse system configurations of the os.conf file. For details about how to write a script, see *attentions on script parsing* and *parsing script example*.

Example of the os.conf file

For instances using classic networks

```
hostname = iZ23r29djm  jZ
password = cXdlcjEyMz  QK
eth0_ip_ad  dr = 10 . 171 . 254 . 123
eth0_mac_a  ddr = 00 : 8c : fa : 5e : 14 : 23
eth0_netma  sk = 255 . 255 . 255 . 0
eth0_gatew  ay = 10 . 171 . 254 . 1
eth0_route =" 10 . 0 . 0 . 0 / 8   10 . 171 . 254 . 1 ; 172 . 16 . 0
. 0 / 12   10 . 171 . 254 . 1 "
eth1_ip_ad  dr = 42 . 120 . 74 . 105
eth1_mac_a  ddr = 00 : 8c : fa : 5e : 14 : 24
eth1_netma  sk = 255 . 255 . 255 . 0
eth1_gatew  ay = 42 . 120 . 74 . 1
eth1_route =" 0 . 0 . 0 . 0 / 0   42 . 120 . 74 . 1 "
dns_namese  rver =" 7 . 7 . 7 . 7   8 . 8 . 8 . 8 "
```

The following table describes the parameters.

Parameter	Parameter description
hostname	The host name
password	The password, which is Base64-encoded
eth0_ip_addr	The IP address of the eth0 NIC
eth0_mac_addr	The MAC address of the eth0 NIC
eth0_netmask	The network mask of the eth0 NIC
eth0_gateway	The default gateway of the eth0 NIC
eth0_route	The eth0 intranet route list, in which routes are separated by semicolons (;) by default
eth1_ip_addr	The IP address of the eth1 NIC
eth1_mac_addr	The MAC address of the eth1 NIC
eth1_netmask	The network mask of the eth1 NIC
eth1_gateway	The default gateway of the eth1 NIC
eth1_route	The eth1 internet route list, in which routes are separated by semicolons (;) by default
dns_nameserver	The DNS address list, in which addresses are separated by spaces by default

For instances using VPCs

```
hostname = iZ23r29djm  jZ
password = cXdlcjEyMz  QK
eth0_ip_ad  dr = 10 . 171 . 254 . 123
eth0_mac_a  ddr = 00 : 8c : fa : 5e : 14 : 23
eth0_netma  sk = 255 . 255 . 255 . 0
eth0_gatew  ay = 10 . 171 . 254 . 1
eth0_route =" 0 . 0 . 0 . 0 / 0   10 . 171 . 254 . 1 "
dns_namese  rver =" 7 . 7 . 7 . 7   8 . 8 . 8 . 8 "
```

The following table describes the parameters.

Parameter	Parameter description
hostname	The host name
password	The password, which is Base64-encoded
eth0_ip_addr	The IP address of the eth0 NIC
eth0_mac_addr	The MAC address of the eth0 NIC
eth0_netmask	The network mask of the eth0 NIC
eth0_gateway	The default gateway of the eth0 NIC

Parameter	Parameter description
eth0_route	The eth0 intranet route list, in which routes are separated by semicolons (;) by default
dns_nameserver	The DNS address list, in which addresses are separated by spaces by default

Script parsing considerations

In normal cases, when an instance is started for the first time, Alibaba Cloud automatically writes information about configuration items into the os.conf file in the `aliyun_cus tom_image` directory in the root directory of the first partition. To configure a customized Linux image, you must create a pre-defined parsing script in the image. Then, Alibaba Cloud reads configuration information about the instance from the os.conf file to complete instance configuration. The following conditions must be met for script parsing:

- Automatic start: The parsing script should be automatically started. To do so, place the script in the `/ etc / init . d /` directory.
- Configuration item value rules: As described in *example of the os.conf file* , instances using classic networks and those using VPCs differ in rules of the number of configuration items and values of some configuration items.
- Configuration file read path: By default, names of the devices allocated for the first partition vary with types of the instances created for the customized Linux image, including I/O optimization instances and non-I/O optimization instances. Therefore, you are recommended to use `uuid` or `label` to indicate devices in the first partition. Because the user password is a Base64-encoded string, it therefore must be Base64-encoded in the script.
- Network type: When using the parsing script to determine the network type, you can check whether there is `eth1_route` or other eth1-related configuration item. To do so, parse and process the instance accordingly after determining whether it uses a classic network or VPC.
  - Instances using VPCs are configured with Internet routes that are specified by the `eth0_route` parameter in the os.conf file.
  - Instances using classic networks are configured with Internet routes that are specified by the `eth1_route` parameter in the os.conf file, and intranet routes are specified by the `eth0_route` parameter.

- Configuration optimization: Configurations in the os.conf file are executed only once during the instance life cycle. You are recommended to delete the os.conf file after the parsing script is successfully executed. The parsing script does not execute configurations in the os.conf file if it does not read any.
- Customized image processing: When you create a customized image based on the customized Linux image, the script requiring automatic start is also included in the new image. Alibaba Cloud will write os.conf file configurations when the instance is started for the first time. Then, the parsing script immediately executes the configurations upon detection.
- Configuration change processing: When instance configurations are changed through the Alibaba Cloud console or APIs, Alibaba Cloud writes related information into the os.conf file. Then, the parsing script executes the configurations again to issue the changes.

Parsing script example

The following uses a parsing script used for CentOS as an example. You can change the script content as needed. Make sure that the script has been successfully debugged in the image before you use the script.

```
#! / bin / bash

###  BEGIN   INIT   INFO
#  Provides :           os - conf
#  Required - Start :    $ local_fs  $ network  $ named  $ remote_fs
#  Required - Stop :
#  Should - Stop :
#  Default - Start :      2   3   4   5
#  Default - Stop :       0   1   6
#  Short - Descriptio  n :  The   initial   os - conf   job ,  config
   the   system .
###  END   INIT   INFO

 first_part  ition_dir ='/ boot /'
 os_conf_di  r =${ first_part  ition_dir }/ aliyun_cus  tom_image
 os_conf_fi  le =${ os_conf_di  r }/ os . conf

 load_os_co  nf () {
  if  [[ - f  $ os_conf_fi  le  ]];  then
  . $ os_conf_fi  le
   return   0
  else
   return   1
  fi
}

 cleanup () {
 #  ensure  $ os_conf_fi  le   is   deleted ,  to   avoid   repeating
   config   system
  rm  $ os_conf_fi  le  >& / dev / null
 #  ensure  $ os_conf_di  r   is   exitst
```

```
  mkdir  - p  $ os_conf_di  r
}

 config_pas  sword () {
  if  [[ - n  $ password  ]];  then
   password =$( echo  $ password  |  base64  - d )
   if  [[ $? ==  0  && - n  $ password  ]];  then
    echo  " root :$ password " |  chpasswd
   fi
  fi
}

 config_hos  tname () {
  if  [[ - n  $ hostname  ]];  then
   sed  - i  " s /^ HOSTNAME =. */ HOSTNAME =$ hostname /" / etc /
 sysconfig / network
   hostname  $ hostname
  fi
}

 config_dns () {
  if  [[ - n  $ dns_namese  rver  ]];  then
   dns_conf =/ etc / resolv . conf
   sed  - i  '/^ nameserver .*/ d ' $ dns_conf
   for  i  in  $ dns_namese  rver ;  do
    echo  " nameserver  $ i " >> $ dns_conf
   done
  fi
}

 is_classic  _network () {
 #  vpc :  eth0
 #  classic :  eth0  eth1
  grep  - q  ' eth1 ' $ os_conf_fi  le
}

 config_net  work () {
 / etc / init . d / network  stop
  config_int  erface  eth0  ${ eth0_ip_ad  dr } ${ eth0_netma  sk }
 ${ eth0_mac_a  ddr }
  config_rou  te  eth0  ${ eth0_route }
  if  is_classic  _network  ;  then
   config_int  erface  eth1  ${ eth1_ip_ad  dr } ${ eth1_netma  sk }
 ${ eth1_mac_a  ddr }
   config_rou  te  eth1  ${ eth1_route }
  fi
 / etc / init . d / network  start
}

 config_int  erface () {
  local  interface =$ 1
  local  ip =$ 2
  local  netmask =$ 3
  local  mac =$ 4
  inteface_c  fg ="/ etc / sysconfig / network - scripts / ifcfg -${
 interface }"
  cat  <<  EOF  > $ inteface_c  fg
 DEVICE =$ interface
 IPADDR =$ ip
 NETMASK =$ netmask
 HWADDR =$ mac
 ONBOOT = yes
 BOOTPROTO = static
 EOF
```

```
}

 config_def  ault_gatew  ay () {
  local   gateway =$ 1
  sed  - i  " s /^ GATEWAY =. */ GATEWAY =$ gateway /" / etc /
 sysconfig / network
}

 config_rou  te () {
  local   interface =$ 1
  local   route =$ 2
  route_conf =/ etc / sysconfig / network - scripts / route -${
 interface }
 > $ route_conf
  echo  $ route  |  sed  ' s /;/\ n /' | \
   while   read   line ;  do
    dst =$( echo  $ line  |  awk  '{ print  $ 1 }')
    gw =$( echo  $ line  |  awk  '{ print  $ 2 }')
    if  !  grep  - q  "$ dst " $ route_conf   2 > / dev / null ;
 then
     echo  "$ dst   via  $ gw   dev  $ interface " >> $ route_conf
    fi
    if  [[ "$ dst " == " 0 . 0 . 0 . 0 / 0 " ]];  then
     config_def  ault_gatew  ay  $ gw
    fi
   done
}

################## sysvinit service portal ####################

 start () {
  if   load_os_co  nf  ;  then
   config_pas  sword
   config_net  work
   config_hos  tname
   config_dns
   cleanup
   return   0
  else
   echo  " not   load  $ os_conf_fi  le "
   return   0
  fi
}

 RETVAL = 0

 case  "$ 1 "  in
     start )
         start
         RETVAL =$?
    ;;
    *)
         echo  " Usage : $ 0  { start }"
         RETVAL = 3
    ;;
 esac
```

```
exit $RETVAL
```

3.3.6 Convert image file format

Only image files in qcow2, RAW, or VHD format can be imported. If you want to import images in other formats, you need to convert the format before importing the image. This topic describes how to use the qemu-img tool to convert other image file formats to VHD or RAW. Using qemu-img, you can convert RAW, qcow2, VMDK, VDI, VHD (vpc), VHDX, qcow1, or QED, to VHD, or implement conversion between RAW and VHD.

Windows

To install qemu-img and convert the image file format, follow these steps:

1. Log on to your server or VM, download *qemu-img* and complete the installation. Installation path: `C:\Program Files\qemu`.
2. Perform the following actions to create an environment variable for qemu-img:
   a. Choose Start > Computer, then right click Properties.
   b. In the left-side navigation pane, click Advanced System Settings.
   c. In the System Properties dialog box, click the Advanced tab, and then click Environment Variables.
   d. In the Environment Variables dialog box, find the Path variable in the System Variables part, and then click Edit. If the Path variable does not exist, click New.
   e. Add a system variable value:
      - In the case of Edit System Variable: In the Variable Value field, add `C:\Program Files\qemu`. Different variable values are separated with semicolon (;).
      - In the case of New System Variable: In the Variable Name field, enter `Path`. In the Variable Value field, enter `C:\Program Files\qemu`.
3. Open Command Prompt in Windows and run the `qemu-img --help` command. If the result is displayed correctly, the environment variable is configured successfully.
4. In the Command prompt, run the `cd [directory of the source image file]` command to change the directory. For example, `cd D:\ConvertImage`.

5. Run the `qemu-img convert -f qcow2 -O raw centos.qcow2 centos.raw` command to convert the image file format. Where:
   - `-f` is followed by the source image format.
   - `-O` (uppercase is required) is followed by the converted image format, the source file name, and the target file name.

   When the conversion is complete, the target file appears in the directory where the source image file is located.

Linux

To install qemu-img and convert the image file format, follow these steps:

1. Install qemu-img, for example:
   - For Ubuntu, run the command: `apt install qemu-img`.
   - For CentOS, run the command: `yum install qemu-img`.
2. Run the `qemu-img convert -f qcow2 -O raw centos.qcow2 centos.raw` command to convert the image file format. Where:
   - `-f` is followed by the source image format.
   - `-O` (uppercase is required) is followed by the converted image format, the source file name, and the target file name.

   When the conversion is complete, the target file appears in the directory where the source image file is located.

Troubleshooting

If errors occur during qemu-img installation and there are no clear prompts about the missing dependent libraries, run `pip install -r requirements.txt` to install all the dependent libraries based on the libraries shown in the file requirements.txt of cloud-init.

Next step

Import custom images

3.3.7 Import custom images

You can import on-premises image files to your ECS environment to create ECS instances or change system disks

Note:

- The time it takes to import an image depends on the size of the image file and the number of concurrent tasks.
- When you import an image, a snapshot is automatically generated. You can view the snapshot information on the Snapshots page in the ECS Console. Before the import image task is completed, the status of the snapshot is displayed as Failed. When the task is completed, the status is automatically updated to Successful. The snapshot capacity is the size of the imported image file, regardless of the system disk size that was set when the image was imported.

Prerequisites

Before importing an image, we recommend that you:

- Review the *notes for importing images*, *customize Linux images*, and *convert image format* to understand the limitations of importing an on-premises image.
- *Activate OSS*.
- (Optional) If you are using a RAM sub-account, you need to contact the master account in advance to obtain the permission for the `AliyunECSI mageImport DefaultRol e` role.

Procedure

To import custom images in the ECS console, follow these steps:

1. Use an OSS third-party client, OSS API or OSS SDK to upload the prepared custom image. If the file you want to upload is larger than 5 GiB, see *multipart upload*.
2. Log on to the *ECS console*.
3. In the left-side navigation pane, choose Snapshots and Images > Images.
4. Click Import Image.
5. In the Import Image dialog box, click Confirm Address as follows.
6. In the Cloud Resource Access Authorization window, select `AliyunECSI mageImport DefaultRol e` and `AliyunECSE xportDefau ltRole`, then

click Confirm Authorization Policy to allow the ECS service to access your OSS resources.

7. On the Images page, click Import Image again.
8. In the Import Image dialog box, enter the following information:
   - Region of Image: Select the region where the OSS Bucket of the image file to upload is located.
   - OSS Object Address: Copy the object address of the image file from the OSS console. For more information, see *download an object*.
   - Image Name: Enter a name for the custom image. The name must be 2 to 128 characters in length and can contain letters, numbers, Chinese characters, periods (.), underscores (_), colons (:), and hyphens (-).
   - Operating System: Select Windows or Linux, that is, the same as that of your image. If you want to import a non-standard platform image, select Linux.
   - System Disk Size: The system disk size, which ranges from 40 GiB to 500 GiB.
   - System Architecture: Choose x86_64 for 64 bit operating systems and choose i386 for 32 bit operating systems.
   - Platform: The options depend on the Operating System you chose.
     - Windows: Windows Server 2003, Windows Server 2008, and Windows Server 2012.
     - Linux: Centos, SUSE, Ubuntu, Debian, FreeBSD, CoreOS, Aliyun, Customized Linux, and Others Linux (*open a ticket* to confirm the selected edition is supported).
     - If your image OS is a custom edition developed from Linux kernel, *open a ticket* to contact us.
   - Image Format: Supports qcow2, RAW, and VHD. Qcow2 or VHD is recommended.
   - Image Description: Enter a description of the custom image.
   - Add Images of Data Disks: Choose this option if you want to import an image that contains data disks. Supported data disk capacity ranges from 5 GiB to 2,000 GiB.
9. Click OK.

10.(Optional) You can view the task progress in the image list of the import region. Before the task is completed, you can find the imported custom image through *Tasks* management, and, if needed, cancel the import task.

You can also use the ECS API *ImportImage* to import a custom image.

Next step

Create an instance from a custom image.

References

- *Custom images FAQ*
- *Create and import on-premise images by using Packer*

3.4 Copy images

Copying images allows you to deploy an application across regions that runs the same image environment. You can copy a custom image from one region to another. The task completion time depends on the network transfer speed and the number of concurrent tasks in the queue.

Attention

- Upon copying a custom image, a corresponding snapshot is created in the target region. The image is then created from that snapshot in the target region. As a result, fees are calculated due to data transfer between different regions. However, no fee is charged for such traffic. The billing policy is subject to change.
- After copying a custom image, an identical image is created in the target region. However, the relevant role and service authorization information is lost, which is also true for previously configured *user data*.

Procedure

To copy images in the ECS console, follow these steps:

1. Log on to the *ECS console*.
2. Select the target region.
3. In the left-side navigation pane, select Snapshots and Images > Images.

4. Select the custom image you want to copy. Note that Type must be Custom Images. Then, in the Actions column, click Copy Image.

   Note:
   If your custom image is larger than 200 GiB, when you click Copy Image you will be directed to open a ticket to complete the operation.

5. In the Copy Image dialog box, you can find the ID of the selected image. Complete the following configurations:

   a. Select the Target Region.
   b. Enter the Custom Image Name and Custom Image Description that are shown in the target region.
   c. Click OK.

6. Switch to the target region and check the progress. When 100% is displayed, the image is copied successfully.

   Note:
   If Progress is not 100%, Status is Creating. In this case, you can click Cancel Copy to cancel the operation. After the operation is canceled, the image information is removed from the target region.

You can also use the ECS APIs *#unique_80* and *#unique_81* to perform the operation.

Additional operations

When an image's status is Available, you can use it to *create an instance* or *change the system disk*.

You can also view all copied snapshot in the target region.

FAQ

FAQs about copying images

3.5 Share images

After creating a custom image, you can share it with other Alibaba Cloud users. Shared images help new users adapt to ECS faster as they can quickly create ECS instances and set up business environments based on your custom images. Moreover,

shared images do not consume the image quota of the account from which an image is shared.

Attention

You can only share custom images you have created, not custom images created and shared by other users. Each custom image can be shared with up to 50 users within the same Alibaba Cloud region. That is, an image cannot be shared across regions.

Before sharing an image, make sure that all sensitive data and files have been deleted from the image.

> **Note:**
> The integrity or security of shared images is not guaranteed. Make sure that you use only images shared by trusted accounts before using shared images. Besides, you shall bear the risk on your own. After you create an instance based on a shared image, be sure to *connect the instance* to check the integrity and security of the image.

Sharing image restrictions

If your custom image has been shared with other accounts, you must remove all the sharing relationships for that image before you can delete the image. After deleting a shared custom image:

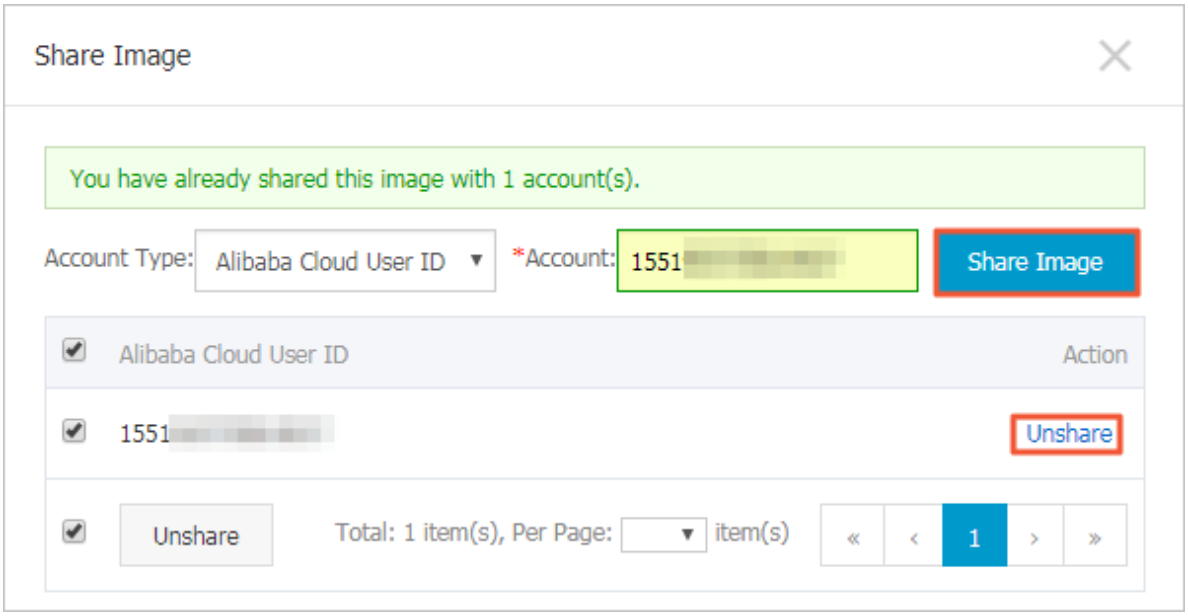
- Users who are using the shared image will no longer be able to find the image through the ECS console or ECS API, nor can they use the image to create ECS instances or replace system disks.
- ECS instances that are created from the shared image cannot re-initialize their system disks.

Share an image

To share an image in the ECS console, follow these steps:

1. Log on to the *ECS console*.
2. Select the target region.
3. In the left-side navigation pane, choose Snapshots and Images > Images.
4. Select the target Custom Image the, in the Actions column, click Share Image.

5. In the pop-up dialog box, select Alibaba Cloud Account ID in the Account Type drop-down list. Then, enter the account ID that you want to share the image with in the Account box. For more information, see *Appendix:How to get the account ID?*.



> **Note:**
> If you want to stop sharing the image with an account, click Unshare next to the account. After you cancel the sharing, that account will be unable to query and use the image. This means that i that account has already created an instance by using this shared image, the instance will be unable to *re-initialize the system disk*.

6. (Optional) For the accounts with whom you share an image, these account can view the shared image in Snapshots and Images > Images > Share Image in the same region in the ECS console.

You can also use the ECS APIs `ModifyImageSharePermission` and `DescribeImageSharePermission` to share an image.

Next steps

After an image is shared with other users, they can use it to create one or more instances.

1. Log on to the *ECS console*.
2. Create one or more instances by referring to *Step 2. Create an instance*Create an instance in *Quick Start*. Note that you should select Shared Image during the procedure.

They can also use the shared image to *#unique_29* for instances.

Appendix: How to get the account ID?

To find your account ID, follow these steps:

1. Log on to the ECS console.
2. Hover your mouse over your avatar and then click Security Settings from the account menu.
3. On the page that appears, the account ID is displayed at the right as follows.

3.6 Export custom images

You can export custom images for on-premises testing or for Apsara stack environments.

> Note:
>
> - The time it takes to export an image depends on the size of the image file and the number of export tasks in the queue.
> - Exported images are stored in your *OSS* bucket. This means you are billed for the OSS storage and download traffic. For more information, see OSS *billing items*.

Limitations

Currently, the image export function has the following limitations:

- You cannot export custom images that are created by a system disk snapshot from the *Alibaba Cloud Marketplace*.
- You can export the custom images that contain four snapshots of data disks at most , and for a single data disk, the maximum volume must be no greater than 500 GiB.
- When using exported custom images to *create an instance by using the wizard*, you must confirm that the file device recorded in /etc/fstab corresponds to the exported data disk snapshot information.

Prerequisites

Before exporting a custom image, you must:

- *Open a ticket* to activate the image export feature, and describe the use cases of the exported images in the ticket.

- Activate OSS and make sure that the region where your custom images are located has an available OSS bucket. For more information, see *create a bucket*.

Procedure

To export a custom image in the ECS console, follow these steps:

1. Log on to the *ECS console*.
2. In the left-side navigation pane, choose Snapshot & Images > Images.
3. Select the target region.
4. Find the custom image you want to export and then, in the Actions column, click Export Image.
   a. In the Export Image dialog box, click Conform Address.
   b. In the Cloud Resource Access Authorization window, click Confirm Authorization Policy to allow ECS to access your OSS resources.
5. Return to the ECS console homepage. In the Actions column of the Images page, click Export Image again.
6. In the Export Image dialog box:
   - Select the OSS bucket in the specified region.
   - Set the prefix of the object name of the exported image. For example, if you set Demo as the prefix, then the exported image file displayed in the OSS bucket is named Demo-[automatically generated file name].
7. Click OK.
8. (Optional) Cancel the image export task. Before the task is completed, you can go to the *Tasks* management page in the ECS console, find the relevant task in the specified region and cancel the task.

You can also use the ECS APIs *ExportImage* and *CancelTask* to perform the preceding operations.

Next steps

When an exported custom image contains a data disk snapshot, multiple files appear in your OSS. The file name with `system` indicates a system disk shapshot and the file name with `data` indicates a data disk snapshot. A data disk snapshot has an identifier corresponding to the data disk, which is the mount point of the data disk, such as xvdb or xvdc.

1. Log on to the *OSS console* to query the export result.
2. After the custom image is exported successful, *download the object* and then download the custom image file.

> Note:
> The format of the image file is RAW by default. However, the .tar.gz format is also available during the gated launch period, and the file is in the .raw format after you unzip the .tar.gz file. If you are using Mac OS X operating system, the agent gnu-tar is recommended to unzip the file.

4 Marketplace images

An Alibaba Cloud Marketplace image is equivalent to the installation disk for an Elastic Compute Service (ECS) instance. A Marketplace image allows you to quickly obtain a running environment for ECS instances and any pre-installed software applications. Such an image can be used for site deployment, application development, and visualized management. Marketplace images effectively allow ECS instances to be used out-of-the-box, helping to reduce costs.

Select a Marketplace image when creating an instance

We recommend that you use a Marketplace image if you are new to working with ECS instances. To deploy a Marketplace image, follow these steps:

1. Go to the *ECS purchase* page.
2. Select and configure your image. For more information, see *create an Instance*. Then, on the Image configuration page, choose Marketplace Image > Select from image market (including operating system).

Purchase an image from Alibaba Cloud Marketplace and create an instance

1. Go to *Alibaba Cloud Marketplace*.
2. Select the required image and click Buy Now.
3. You may be required to log on to the Alibaba Cloud console before proceeding.
4. Select and configure your image. For more information, see *create an instance*.

Change the operating system by using the Marketplace image

If you have purchased ECS instances, use an image to deploy the running environmen t or install software applications as follows:

Note:

If you change the image, the data on the system disk will be lost. Therefore, we recommend that you back up your data before changing your operating system. For more information, see *Create snapshots*.

1. Log on to the *ECS console*.
2. Stop the target instance.
3. On the Replace System Disk page, select Marketplace Image in the Image Type setting. For more information, see *replace the system disk (non-public image)*.

5 Open source tools

6 Change the operating system

You can convert the OS running on your ECS instance to another supported OS through the ECS console.

To change the operating system, you must change the system disk of an instance:

- If you want to use a custom image, see *change the system disk (custom image)*.
- If you want to use a public image, see *change a system disk (public image)*.

Note:

Currently, instances that are hosted in regions outside of mainland China do not support swapping between Linux and Windows OSs. If your instance is hosted in one of these regions, you can only change its version of Windows OS to another version of Windows, or replace its current Linux OS with another Linux OS.

7 FAQ

7.1 Image FAQ

Does the system disk of an ECS instance support KMS encryption?

Yes. The system disk of an ECS instance supports KMS encryption by using the key that is automatically generated by KMS. Currently, support for BYOK encryption is in development.

Is KMS encryption supported by Terraform or Packer?

In Terraform, you can set the `encrypted` parameter to enable or disable KMS encryption. For more information, see *Argument Reference*. Currently, support for KMS encryption in Packer is in development.