

**Elastic Volume Service**

# Getting Started

**Issue**            01  
**Date**             2023-07-20



**Copyright © Huawei Technologies Co., Ltd. 2023. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

## **Trademarks and Permissions**



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

## **Notice**

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

---

# Contents

---

<b>1 Process Overview.....</b>	<b>1</b>
<b>2 Quickly Creating and Using an EVS Disk.....</b>	<b>3</b>
<b>3 Step 1: Preparations.....</b>	<b>4</b>
<b>4 Step 2: Purchase an EVS Disk.....</b>	<b>5</b>
<b>5 Step 3: Attach an EVS Disk.....</b>	<b>13</b>
5.1 Attaching a Non-Shared Disk.....	13
5.2 Attaching a Shared Disk.....	15
<b>6 Step 4: Initialize an EVS Data Disk.....</b>	<b>19</b>
6.1 Introduction to Data Disk Initialization Scenarios and Partition Styles.....	19
6.2 Initializing a Linux Data Disk (fdisk).....	21
6.3 Initializing a Linux Data Disk (parted).....	27
6.4 Initializing a Linux Data Disk Larger Than 2 TiB (parted).....	33
<b>A Change History.....</b>	<b>40</b>

# 1 Process Overview

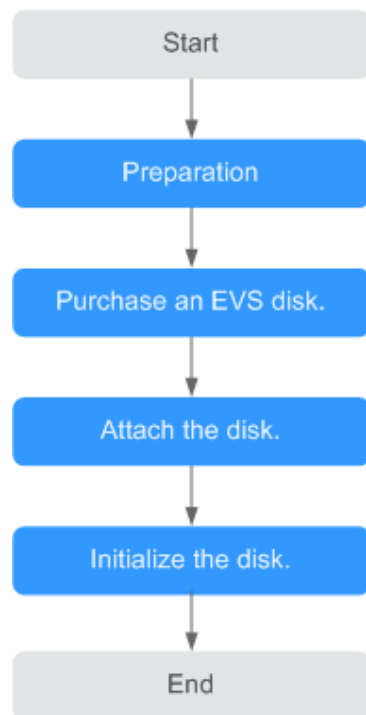
EVS disks can be attached to servers and be used as system disks or data disks. [Table 1-1](#) lists the disk purchasing methods.

**Table 1-1** Method of purchase

Disk	Description	Method
System disk	System disks are purchased along with servers and cannot be purchased separately.	<ul style="list-style-type: none"><li>• <a href="#">Purchasing an ECS</a></li><li>• <a href="#">Creating a BMS</a></li></ul>
Data disk	Data disks can be purchased along with servers or separately.	<ul style="list-style-type: none"><li>• <a href="#">Purchasing an ECS</a></li><li>• <a href="#">Creating a BMS</a></li><li>• <a href="#">Purchase an EVS Disk</a></li></ul>

[Figure 1-1](#) shows how to purchase a data disk separately.

**Figure 1-1** Process overview



1. **Preparations:** Register with Huawei Cloud. For details, see [Preparations](#).
2. **Purchase a disk.** Configure the disk parameters, including the disk type, size, name, and other information. For details, see [Purchase an EVS Disk](#).
3. **Attach the data disk.** Attach the separately purchased disk to a server. For details, see the following sections:
  - [Attaching a Non-Shared Disk](#)
  - [Attaching a Shared Disk](#)
4. **Initialize the data disk.** Log in to the server and initialize the data disk before using it. For details about how to initialize the disk, see the following sections:
  - [Introduction to Data Disk Initialization Scenarios and Partition Styles](#)
  - Linux
    - [Initializing a Linux Data Disk \(fdisk\)](#)
    - [Initializing a Linux Data Disk \(parted\)](#)
    - [Initializing a Linux Data Disk Larger Than 2 TiB \(parted\)](#)

# 2 Quickly Creating and Using an EVS Disk

---

# 3 Step 1: Preparations

---

Before using the EVS service, you need to make the following preparations:

- [Registering with Huawei Cloud](#)

## Registering with Huawei Cloud

If you already have a Huawei Cloud account, skip this part. If you do not have a Huawei Cloud account, perform the following steps to create one:

1. Visit <https://support.huaweicloud.com/eu/> and click **Register**.

# 4 Step 2: Purchase an EVS Disk

## Scenarios

EVS disks can be used as system disks or data disks for servers. You can purchase data disks on the EVS console, or purchase them together with system disks on the cloud server console.

This section describes how to purchase data disks on the EVS console.

## Constraints

**Table 4-1** Constraints on purchasing disks

Purchased On	Description
The EVS console	<ul style="list-style-type: none"><li>• Disks purchased on the EVS console are data disks. You need to manually attach them to servers.</li><li>• Disks can only be attached to servers in the same region and AZ. Once purchased, the region and AZ cannot be changed.</li><li>• Yearly/Monthly disks can be purchased on the EVS console only when you have selected to attach them to a yearly/monthly server.</li><li>• There are quantity and capacity quotas on EVS disks, so properly plan the number of disks and total disk capacity your workloads require. For details, see <a href="#">Managing Quotas</a>.</li></ul>
The Cloud Server Console	<ul style="list-style-type: none"><li>• System disks can only be purchased together with servers and are automatically attached.</li><li>• Data disks purchased together with servers or added after the server purchase are automatically attached.</li><li>• Disks will have the same billing mode as their server if the disks are purchased together with the server.</li><li>• By default, disks purchased with ECSs are VBD disks, and those purchased with BMSs are SCSI disks.</li></ul>



Purchased On	Description
-	Capacities of multiple disks cannot be combined, and the capacity of a single disk cannot be split.

## Procedure

- Step 1** Log in to the [management console](#).
- Step 2** Under **Storage**, click **Elastic Volume Service**.  
The disk list page is displayed.
- Step 3** Click **Buy Disk**.
- Step 4** Configure basic disk information according to [Table 4-2](#).

**Table 4-2** Disk parameters

Parameter	Sub-Parameter	Description	Example Value
Billing Mode	-	Mandatory Pay-per-use	Pay-per-use
Region	-	Mandatory Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.	-
AZ	-	Mandatory The availability zone (AZ) where you want to create the disk. <b>NOTE</b> <ul style="list-style-type: none"><li>Disks can only be attached to the servers in the same AZ.</li><li>The AZ of a disk cannot be changed after the disk has been created.</li></ul>	AZ1

Parameter	Sub-Parameter	Description	Example Value
Disk Specifications	Disk Type	<p>Mandatory</p> <p>The available disk types are as follows:</p> <ul style="list-style-type: none"> <li>• High I/O</li> <li>• General Purpose SSD</li> <li>• Ultra-high I/O</li> <li>• Extreme SSD</li> </ul> <p><b>NOTE</b> When a disk is created from a snapshot, the disk type of the new disk will be consistent with that of the snapshot's source disk.</p>	Ultra-high I/O
	Disk Size (GiB)	<p>Mandatory</p> <p>The disk size. Only data disks can be created on the current page, and the disk size ranges from 10 GiB to 32,768 GiB.</p> <p><b>NOTE</b></p> <ul style="list-style-type: none"> <li>• When you use a backup to create a disk, the disk capacity must be greater than or equal to the backup size. In the condition that you do not specify the disk capacity, if the backup size is smaller than 10 GiB, the default capacity 10 GiB will be used as the disk capacity; if the backup size is greater than 10 GiB, the disk capacity will be consistent with the backup size.</li> <li>• When you use a snapshot to create a disk, the disk capacity must be greater than or equal to the snapshot size. In the condition that you do not specify a disk capacity, if the snapshot size is smaller than 10 GiB, the default 10 GiB will be used as the disk capacity; if the snapshot size is greater than 10 GiB, the disk capacity will be consistent with the snapshot size.</li> <li>• The system shows you the maximum number of disks as well as the maximum disk capacity allowed to purchase. To ensure effective resource usage, if the disk capacity you need exceeds the upper limit, click <b>Increase Quota</b> to obtain a higher quota. You can purchase the disk capacity you need after the request is approved.</li> </ul>	100 GiB

Parameter	Sub-Parameter	Description	Example Value
	<p>Create from</p> <ul style="list-style-type: none"> <li>• Backup</li> <li>• Snapshot</li> <li>• Image</li> </ul>	<p>Optional</p> <ul style="list-style-type: none"> <li>• If you choose <b>Create from Backup</b>, the backup data is used to create the disk. Click <b>Create from</b> and choose <b>Backup</b>. On the displayed page, select the target backup and click <b>OK</b>. <b>NOTE</b> <ul style="list-style-type: none"> <li>- One backup cannot be used for concurrent disk creation operations at the same time. For example, if you are creating disk A from a backup, this backup can be used to create another disk only after disk A has been created.</li> <li>- If a disk is created from a backup of a system disk, the new disk can be used as a data disk only.</li> </ul> </li> <li>• If you choose <b>Create from Snapshot</b>, the snapshot data is used to create the disk. Click <b>Create from</b> and choose <b>Snapshot</b>. On the displayed page, select the target snapshot and click <b>OK</b>. <b>NOTE</b> <ul style="list-style-type: none"> <li>- The device type of the new disk is the same as that of the snapshot's source disk.</li> <li>- The encryption attribute of the new disk is the same as that of the snapshot's source disk.</li> </ul> </li> <li>• If you choose <b>Create from Image</b>, the image data is used to create the disk. Click <b>Create from</b> and choose <b>Image</b>. On the displayed page, select the target image and click <b>OK</b>. <b>NOTE</b> <ul style="list-style-type: none"> <li>- The device type of the new disk is the same as that of the image's source disk.</li> <li>- The encryption attribute of the new disk is the same as that of the image's source disk.</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Create from Backup: autobackup-001</li> </ul>

Parameter	Sub-Parameter	Description	Example Value
More	Advanced Settings <ul style="list-style-type: none"> <li>• Share</li> <li>• SCSI</li> <li>• Encryption</li> </ul>	<p>Optional</p> <ul style="list-style-type: none"> <li>• Share If you select <b>Share</b>, a shared disk is created. A shared disk can be attached to up to 16 servers. If you do not select <b>Share</b>, a non-shared disk is created, and the disk can be attached to one a server only.</li> </ul> <p>If you select both <b>SCSI</b> and <b>Share</b>, a shared SCSI disk is created.</p> <p><b>NOTE</b> The sharing attribute of a disk cannot be changed after the disk has been created.</p> <ul style="list-style-type: none"> <li>• SCSI If you select <b>SCSI</b>, a SCSI disk is created. Such disks allow the server OS to directly access the underlying storage media and send SCSI commands to the disks. If you do not select <b>SCSI</b>, a VBD disk is created. That said, the disk device type is VBD, the default device type.</li> </ul> <p><b>NOTE</b> The device type of a disk cannot be changed after the disk has been created.</p> <ul style="list-style-type: none"> <li>• Encryption The encryption function on this console is used for data disk encryption only.</li> </ul> <p>System disk encryption relies on the image. For details, see <a href="#">Creating Encrypted Images</a>.</p> <p>To use the disk encryption function, select <b>Encryption</b>. Confirm to create the agency on the displayed <b>Create Agency</b> dialog box. Then configure the encryption parameters as prompted:</p> <ul style="list-style-type: none"> <li>- Create Agency An agency is a trust relationship between two tenants or services. A tenant can create an agency to grant resource access rights to another tenant or service. If the</li> </ul>	-

Parameter	Sub-Parameter	Description	Example Value
		<p>KMS access rights are not granted to EVS, the <b>Create Agency</b> dialog box will be displayed. Otherwise, it will not be displayed.</p> <p>Click <b>Yes</b> to grant the KMS access rights to EVS. After the rights have been granted, EVS can obtain KMS keys to encrypt or decrypt EVS disks.</p> <p>After the KMS access rights have been granted, follow-up operations do not require the rights to be granted again.</p> <ul style="list-style-type: none"> <li>- KMS Key Name <b>KMS Key Name</b> is displayed only after the KMS access rights have been granted.</li> </ul> <p><b>KMS Key Name</b> is the identifier of the key, and you can use <b>KMS Key Name</b> to specify the KMS key that is to be used for encryption. One of the following keys can be used:</p> <p>Default Master Key: After the KMS access rights have been granted to EVS, the system automatically creates a Default Master Key and names it <b>evs/default</b>.</p> <p>An existing or new CMK. For details about how to create one, see <a href="#">Creating a CMK</a>.</p> <p><b>NOTE</b></p> <ul style="list-style-type: none"> <li>- Before you use the encryption function, KMS access rights need to be granted to EVS. If you have the right to grant the permission, grant the KMS access rights to EVS directly. If you do not have this permission, contact a user with the security administrator rights to grant KMS access rights to EVS, then repeat the preceding operations.</li> <li>- The encryption attribute of a disk cannot be changed after the disk has been created.</li> </ul>	

Parameter	Sub-Parameter	Description	Example Value
	Tag	<p>Optional</p> <p>During the EVS disk creation, you can tag the EVS resources. Tags identify cloud resources for purposes of easy categorization and quick search.</p> <p>A tag is composed of a key-value pair.</p> <ul style="list-style-type: none"> <li>• Key: Mandatory if the disk is going to be tagged A tag key can contain a maximum of 36 characters. It consists of letters, digits, underscores (_), hyphens (-), and Unicode characters (\u4E00-\u9FFF).</li> <li>• Value: Optional if the disk is going to be tagged A tag value can contain a maximum of 43 characters. It consists of letters, digits, underscores (_), periods (.), hyphens (-), and Unicode characters (\u4E00-\u9FFF).</li> </ul> <p><b>NOTE</b></p> <ul style="list-style-type: none"> <li>• A maximum of 10 tags can be added for an EVS disk.</li> <li>• Tag keys of the same EVS disk must be unique.</li> </ul> <p>For details about tags, see the <i>Tag Management Service User Guide</i>.</p>	-
Enterprise Project	-	<p>Mandatory</p> <p>When creating EVS disks, you can add the disks to an existing enterprise project.</p> <p>An enterprise project facilitates project-level management and grouping of cloud resources and users. The default project is <b>default</b>.</p>	default

Parameter	Sub-Parameter	Description	Example Value
Disk Name	-	<p>Mandatory</p> <ul style="list-style-type: none"> <li>If you create disks individually, this parameter value is used as the actual disk name. The name can contain a maximum of 64 characters.</li> <li>If you create disks in a batch, this parameter value is used as the prefix of disk names, and one disk name will be composed of this parameter value and a four-digit number. The name can contain a maximum of 59 characters.</li> </ul>	<p>For example, if you create two disks and set <b>volume</b> for <b>Disk Name</b>, the EVS disk names will be <b>volume-0001</b> and <b>volume-0002</b>.</p>
Quantity	-	<p>(Optional) The number of disks to be created. The default value is set to <b>1</b>, which means only one disk is created. Currently, you can create up to 100 disks at a time.</p> <p><b>NOTE</b></p> <ul style="list-style-type: none"> <li>If the disk is created from a backup, batch creation is not possible, and this parameter must be set to <b>1</b>.</li> <li>If the disk is created from a snapshot, batch creation is not possible, and this parameter must be set to <b>1</b>.</li> <li>The system shows you the maximum number of disks as well as the maximum disk capacity allowed to purchase. To ensure effective resource usage, if the number of disks you need exceeds the upper limit, click <b>Increase Quota</b> to obtain a higher quota. You can purchase the disks you need after the request is approved.</li> </ul>	1

**Step 5** Click **Next**.

1. On the **Details** page, check the disk information again.
2. Confirm the information and click **Submit**. The system displays a message indicating request submitted successfully.
3. Click **Back to Disk List** to return to the **Elastic Volume Service** page.

**Step 6** In the disk list, view the disk status.

When the disk status changes to **Available**, the disk is successfully created.

----End

# 5 Step 3: Attach an EVS Disk

---

## 5.1 Attaching a Non-Shared Disk

### Scenarios

Separately purchased EVS disks are data disks. In the disk list, the function of such disks is displayed as **Data disk**, and the status is displayed as **Available**. In this case, you need to attach the data disks to servers for use.

A system disk must be created together with an ECS and is automatically attached. In the disk list, the function of such disks is displayed as **System disk**, and the status is displayed as **In-use**. After a system disk is detached from an ECS, the disk function changes to **Bootable disk**, and the status changes to **Available**.

#### NOTE

Bootable disks are the system disks detached from servers. A bootable disk can be re-attached to a server and be used as a system disk or data disk depending on the disk function selected. For details, see [Attaching an Existing System Disk](#).

This section describes how to attach a non-shared disk.

### Prerequisites

- The non-shared disk status is **Available**.
- The account is not in arrears.

### Constraints

- Cloud servers created from ISO images are only used for OS installation. They have limited functions and cannot have EVS disks attached.
- A non-shared disk can be attached to one server only.
- The disk and the server must be in the same region and AZ.
- A detached, non-shared yearly/monthly data disk purchased together with a server can only be re-attached to the original server and be used as a data disk.



- A shared disk can be attached only when the servers' statuses are **Running** or **Stopped**.
- A frozen disk cannot be attached.
- A detached, yearly/monthly system disk purchased together with a server can be re-attached and used as a data disk for any server. If you want to use it again as a system disk, you must attach it to the original server.

## Attaching the Disk on the EVS Console

**Step 1** Log in to the [management console](#).

**Step 2** Under **Storage**, click **Elastic Volume Service**.

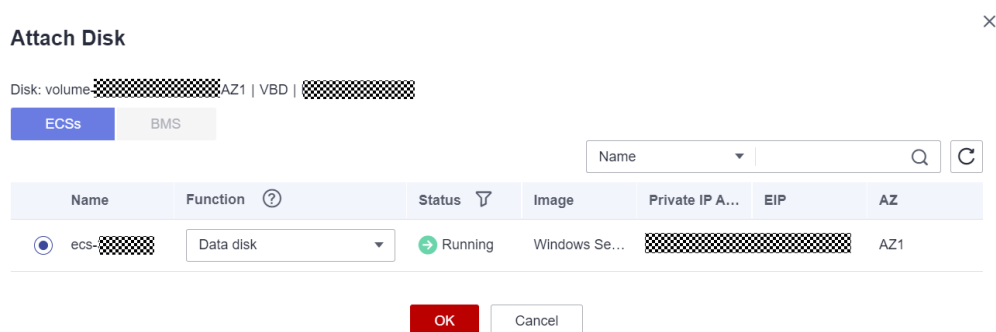
The disk list page is displayed.

**Step 3** Locate the target disk in the list and click **Attach**.

**Step 4** Select the server and then select the disk function from the drop-down list. Ensure that the disk and server are in the same AZ.

One device name can be used for one disk only. For how to obtain the disk name in the OS, see section "How Do I Obtain My Disk Name in the ECS OS Using the Device Identifier Provided on the Console?" in the *Elastic Cloud Server User Guide*.

**Figure 5-1** Attach Disk



**Step 5** Click **OK**.

A dialog box is displayed, showing "The attaching process is NOT completed yet. You must initialize the disk before using it."

**Step 6** Click **OK** to return to the disk list page.




The status of the disk is **Attaching**, indicating that the disk is being attached to the server. When the disk status changes to **In-use**, the disk is successfully attached.

**Step 7** Initialize the disk.

After the disk has been attached to a server, the disk can be used only after you have initialized it. For details, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

----End

## Attaching the Disk on the ECS Console

1. Log in to the management console.
  2. Click  in the upper left corner and select your region and project.
  3. Click . Under **Compute**, choose **Elastic Cloud Server**.
  4. In the search box above the upper right corner of the ECS list, enter the ECS name, IP address, or ID for search.
  5. Click the name of the target ECS.  
The page providing details about the ECS is displayed.
  6. Click the **Disks** tab. Then, click **Attach Disk**.  
The **Attach Disk** dialog box is displayed.
  7. Select the target disk and specify it as the system disk or a data disk.
    - For KVM ECSs, you can specify the disk as the system disk or a data disk but cannot specify a specific device name.
    - For Xen ECSs, you can specify a specific device name, such as `/dev/vdb`.
-  **NOTE**
- If no disks are available, click **Create Disk** in the lower part of the list.
  - For the restrictions on attaching disks, see [What Are the Requirements for Attaching an EVS Disk to an ECS?](#)
8. Click **OK**.  
After the disk is attached, you can view the information about it on the **Disks** tab.

## Follow-Up Operations

If you are attaching a new disk, you must then log in to the server and initialize the disk before it can be used. To learn how to initialize disks, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

## Related Operations

If your disk cannot be attached to a server, see [Why Can't My Disk Be Attached?](#)

If the disk you are going to attach contains data, see [Attaching an Existing Disk](#).

If the attached data disk is not showing up, see [Why Can't I View the Attached Data Disk on the Server?](#)

## 5.2 Attaching a Shared Disk

### Scenarios

Separately purchased shared EVS disks are data disks. In the disk list, the function of such a disk is displayed as **Data disk**, and the status is displayed as **Available**. In this case, you need to attach the shared data disk to servers for use.

If your disk cannot be attached to a server, see [Why Can't My Disk Be Attached?](#)

## Prerequisites

- The shared disk status is **In-use** or **Available**.
- The account is not in arrears.

## Constraints

### NOTICE

If you simply attach a shared disk to multiple servers, files cannot be shared among them. Because there are no mutually agreed data read/write rules among servers, read and write operations from them may interfere with each other, or unpredictable errors may occur. To share files between servers, set up a shared file system or a clustered management system first.

- A shared disk can be attached to a maximum of 16 servers. These servers and the shared disk must be in the same AZ within a region.
- A shared, **In-use** disk can be attached to other servers only when the maximum number of servers that the disk can be attached to has not been reached.
- All the servers attached with a shared disk must run either Windows or Linux. For example, if you attach a shared disk to multiple Windows servers and then detach it from these servers, the shared disk cannot be attached to Linux servers later. This is because Windows and Linux support different file systems and cannot identify the original file system on the disk. Improper operations may damage the original file system.
- A shared disk can only be used as a data disk. It cannot be used as a system disk.
- Cloud servers created from ISO images are only used for OS installation. They have limited functions and cannot have EVS disks attached.
- A shared disk can be attached only when the servers' statuses are **Running** or **Stopped**.
- A frozen disk cannot be attached.
- A detached, yearly/monthly system disk purchased together with a server can be re-attached and used as a data disk for any server. If you want to use it again as a system disk, you must attach it to the original server.

## Attaching the Disk on the EVS Console

**Step 1** Log in to the [management console](#).

**Step 2** Under **Storage**, click **Elastic Volume Service**.

The disk list page is displayed.

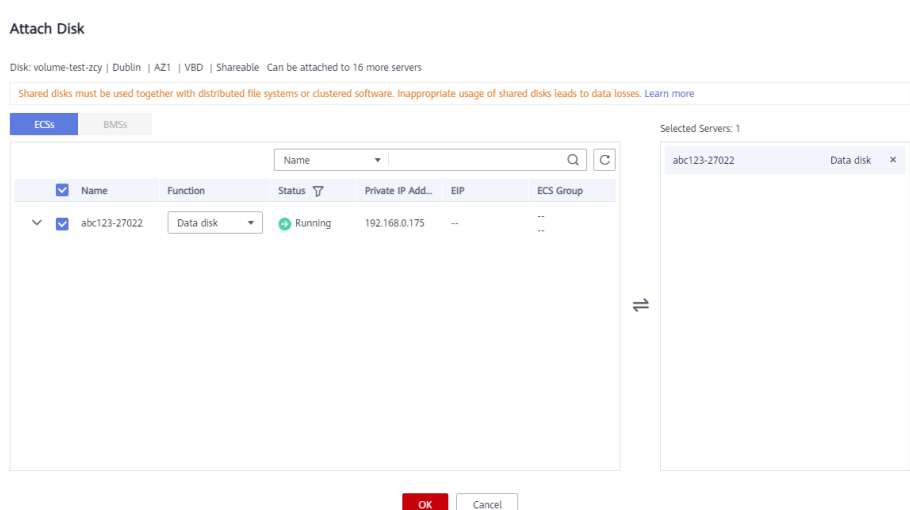
**Step 3** Locate the target disk in the list and click **Attach**.

Shared disks support batch attachment so that you can attach a shared disk to multiple servers. The left area in the **Attach Disk** dialog box shows the server list. After you select the target servers, the selected servers will be displayed in the right area.

**Step 4** Select the target servers you want to attach the shared disk. Ensure that the disk and servers are in the same AZ. After you select the servers, **Data disk** is automatically entered as the disk function for each server.

One device name can be used for one disk only. If a device name has been used, it will no longer be displayed in the drop-down list and cannot be selected.

**Figure 5-2** Attach Disk



**Step 5** Click **OK**.


A dialog box is displayed, showing "The attaching process is NOT completed yet. You must initialize the disk before using it."

**Step 6** Click **OK** to return to the disk list page.

The status of the disk is **Attaching**, indicating that the disk is being attached to the servers. When the disk status changes to **In-use**, the disk is successfully attached.

----End

## Attaching the Disk on the ECS Console

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Under **Compute**, click **Elastic Cloud Server**.
4. In the search box above the upper right corner of the ECS list, enter the ECS name, IP address, or ID for search.
5. Click the name of the target ECS.  
The page providing details about the ECS is displayed.
6. Click the **Disks** tab. Then, click **Attach Disk**.  
The **Attach Disk** page is displayed.
7. Select the target disk and specify it as the system disk or a data disk.

- For Xen ECSs, you can specify a specific device name, such as `/dev/sdb`.
- For KVM ECSs, you can specify the disk as a system disk or data disk but cannot specify a device name.

 **NOTE**

- If no disks are available, click **Create Disk** in the lower part of the list.
  - For the restrictions on attaching disks, see [What Are the Requirements for Attaching an EVS Disk to an ECS?](#)
8. Click **OK**.  
After the disk is attached, you can view the information about it on the **Disks** tab.

## Follow-Up Operations

If you are attaching a new disk, you must then log in to the server and initialize the disk before it can be used. To learn how to initialize disks, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

## Related Operations

If your disk cannot be attached to a server, see [Why Can't My Disk Be Attached?](#)

If the disk you are going to attach contains data, see [Attaching an Existing Disk](#).

If the attached data disk is not showing up, see [Why Can't I View the Attached Data Disk on the Server?](#)

# 6 Step 4: Initialize an EVS Data Disk

---

## 6.1 Introduction to Data Disk Initialization Scenarios and Partition Styles

### Scenarios

After a disk is attached to a server, you need to log in to the server to initialize the disk, that is, format the disk. You must initialize a disk before accessing it.

- System disk  
A system disk does not require manual initialization because it is automatically created and initialized upon server creation. The default partition style is master boot record (MBR).
- Data disk
  - If a data disk is created along with a server, it will be automatically attached to the server.
  - If a data disk is created separately, you need to manually attach it to a server.

In both cases, you must initialize the data disk before using it. Choose an appropriate partition style based on your service plan.

### Prerequisites

- The disk has been attached to a server.
- Data may be lost after a disk is initialized, so you are advised to create [snapshots](#) or [backups](#) before initializing a disk.

### Constraints

- A disk created from a data source does not need to be initialized. Such a disk contains the data of the source in the beginning. Initializing the disk may clear the initial data on it.
- Initializing a disk does not change the server's IP address or the disk ID.
- Initializing a disk does not delete the snapshots created for the disk, so you can still roll back snapshots to the original disk after the disk is initialized.

## Disk Partition Styles

**Table 6-1** lists the common disk partition styles. In Linux, different partition styles require different partitioning tools.

**Table 6-1** Disk partition styles

Disk Partition Style	Maximum Disk Capacity Supported	Maximum Number of Partitions Supported	Linux Partitioning Tool
Master Boot Record (MBR)	2 TiB	<ul style="list-style-type: none"><li>4 primary partitions</li><li>3 primary partitions and 1 extended partition</li></ul> <p>With MBR, you can create several primary partitions and one extended partition. The extended partition must be divided into logical partitions before use. For example, if 6 partitions need to be created, you can create them in the following two ways:</p> <ul style="list-style-type: none"><li>3 primary partitions and 1 extended partition, with the extended partition divided into 3 logical partitions</li><li>1 primary partition and 1 extended partition, with the extended partition divided into 5 logical partitions</li></ul>	<ul style="list-style-type: none"><li>fdisk</li><li>parted</li></ul>
GUID Partition Table (GPT)	18 EiB 1 EiB = 1048576 TiB	Unlimited Disk partitions created using GPT are not categorized.	parted

### NOTICE

The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

If the partition style is changed after the disk has been used, all data on the disk will be lost, so take care to select an appropriate partition style when initializing the disk. If you must change the partition style to GPT after a disk has been used, it is recommended that you back up the disk data before the change.

## Partitioning Operation Guide

For a disk smaller than 2 TiB, see the following sections:

- [Initializing a Linux Data Disk \(fdisk\)](#)
- [Initializing a Linux Data Disk \(parted\)](#)

For a disk larger than 2 TiB, see the following sections:

- [Initializing a Linux Data Disk Larger Than 2 TiB \(parted\)](#)

## 6.2 Initializing a Linux Data Disk (fdisk)

### Scenarios

This section uses CentOS 7.4 64bit to describe how to initialize a data disk attached to a server running Linux and use fdisk to partition the data disk.

The maximum partition size that MBR supports is 2 TiB and that GPT supports is 18 EiB. If the disk size you need to partition is greater than 2 TiB, partition the disk using GPT.

The fdisk partitioning tool is suitable only for MBR partitions, and the parted partitioning tool is suitable for both MBR and GPT partitions. For more information, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

The method for initializing a disk varies slightly depending on the OS running on the server. This document is used for reference only. For the detailed operations and differences, see the product documents of the corresponding OS.

---

#### NOTICE

When using a disk for the first time, if you have not initialized it, including creating partitions and file systems, the additional space added to this disk in an expansion later may not be normally used.

---

### Prerequisites

- A data disk has been attached to a server and has not been initialized.
- You have logged in to the server.
  - For how to log in to an ECS, see [Logging In to an ECS](#).
  - For how to log in to a BMS, see [Logging In to the BMS](#).

### Creating and Mounting a Partition

The following example shows you how a new primary partition can be created on a new data disk that has been attached to a server. The primary partition will be created using fdisk, and MBR will be used. Furthermore, the partition will be formatted using the ext4 file system, mounted on `/mnt/sdc`, and configured to mount automatically at startup.



**Step 1** Query what block devices are available on the server.

### **fdisk -l**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# fdisk -l

Disk /dev/vda: 42.9 GiB, 42949672960 bytes, 83886080 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000bcb4e

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1 *         2048     83886079     41942016   83   Linux

Disk /dev/vdb: 107.4 GiB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

In the command output, this server contains two disks. **/dev/vda** and **/dev/vdb**. **/dev/vda** is the system disk, and **/dev/vdb** is the new data disk.

**Step 2** Launch fdisk to partition the new data disk.

### **fdisk New data disk**

In this example, run the following command:

### **fdisk /dev/vdb**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x38717fc1.

Command (m for help):
```

**Step 3** Enter **n** and press **Enter** to create a new partition.

Information similar to the following is displayed:

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
```

There are two types of disk partitions:

- Choosing **p** creates a primary partition.
- Choosing **e** creates an extended partition.

### **NOTE**

If the MBR partition style is used, a maximum of 4 primary partitions, or 3 primary partitions and 1 extended partition can be created. The extended partition cannot be used directly and must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

**Step 4** Enter **p** and press **Enter** to create a primary partition in this example.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (1-4, default 1):
```

**Partition number** indicates the serial number of the primary partition. The value ranges from **1** to **4**.

- Step 5** Enter the serial number of the primary partition and press **Enter**. Primary partition number **1** is used in this example. One usually starts with partition number **1** when partitioning an empty disk.

Information similar to the following is displayed:

```
Partition number (1-4, default 1): 1
First sector (2048-209715199, default 2048):
```

**First sector** indicates the start sector. The value ranges from **2048** to **209715199**, and the default value is **2048**.

- Step 6** Select the default start sector **2048** and press **Enter**.

The system displays the start and end sectors of the partition's available space. You can customize the value within this range or use the default value. The start sector must be smaller than the partition's end sector.

Information similar to the following is displayed:

```
First sector (2048-209715199, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-209715199, default 209715199):
```

**Last sector** indicates the end sector. The value ranges from **2048** to **209715199**, and the default value is **209715199**.

- Step 7** Select the default end sector **209715199** and press **Enter**.

The system displays the start and end sectors of the partition's available space. You can customize the value within this range or use the default value. The start sector must be smaller than the partition's end sector.

Information similar to the following is displayed:

```
Last sector, +sectors or +size{K,M,G} (2048-209715199, default 209715199):
Using default value 209715199
Partition 1 of type Linux and of size 100 GiB is set
Command (m for help):
```

A primary partition has been created for the new data disk.

- Step 8** Enter **p** and press **Enter** to print the partition details.

Information similar to the following is displayed:

```
Command (m for help): p

Disk /dev/vdb: 107.4 GiB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x38717fc1

   Device Boot      Start         End      Blocks   Id  System
/dev/vdb1          2048    209715199   104856576   83  Linux

Command (m for help):
```

Details about the **/dev/vdb1** partition are displayed.

**Step 9** Enter **w** and press **Enter** to write the changes to the partition table.

Information similar to the following is displayed:

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

The partition is created.

#### NOTE

In case that you want to discard the changes made before, you can exit fdisk by entering **q**.

**Step 10** Synchronize the new partition table to the OS.

**partprobe**

**Step 11** Format the new partition with a desired file system format.

**mkfs -t *File system format* /dev/vdb1**

In this example, the **ext4** format is used for the new partition.

**mkfs -t ext4 /dev/vdb1**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# mkfs -t ext4 /dev/vdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
6553600 inodes, 26214144 blocks
1310707 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2174746624
800 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

The formatting takes a period of time. Observe the system running status and do not exit.

---

#### NOTICE

The partition sizes supported by file systems vary. Choose an appropriate file system format based on your service requirements.

---

**Step 12** Create a mount point.

**mkdir *Mount point***

In this example, the `/mnt/sdc` mount point is created.

**mkdir /mnt/sdc**

 **NOTE**

The `/mnt` directory exists on all Linux systems. If the mount point cannot be created, it may be that the `/mnt` directory has been accidentally deleted. You can run **mkdir -p /mnt/sdc** to create the mount point.

**Step 13** Mount the new partition on the created mount point.

**mount** *Disk partition Mount point*

In this example, the `/dev/vdb1` partition is mounted on `/mnt/sdc`.

**mount /dev/vdb1 /mnt/sdc**

**Step 14** Check the mount result.

**df -TH**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/vda1       ext4      43G   1.9G   39G   5% /
devtmpfs        devtmpfs  2.0G   0   2.0G   0% /dev
tmpfs           tmpfs     2.0G   0   2.0G   0% /dev/shm
tmpfs           tmpfs     2.0G   9.1M   2.0G   1% /run
tmpfs           tmpfs     2.0G   0   2.0G   0% /sys/fs/cgroup
tmpfs           tmpfs     398M   0   398M   0% /run/user/0
/dev/vdb1       ext4     106G   63M  101G   1% /mnt/sdc
```

You should now see that partition `/dev/vdb1` is mounted on `/mnt/sdc`.

 **NOTE**

After the server is restarted, the disk will not be automatically mounted. To configure automount at startup, you will need to modify the `/etc/fstab` file. For details, see [Configuring Automatic Mounting at System Start](#).

----End

## Configuring Automatic Mounting at System Start

The **fstab** file controls what disks are automatically mounted at startup. You can use **fstab** to configure your data disks to mount automatically. This operation will not affect the existing data.

The example here uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names to identify disks in the file because device names are assigned dynamically and may change (for example, from `/dev/vdb1` to `/dev/vdb2`) after a reboot. This can even prevent your disk from booting up.

 **NOTE**

UUID is the unique character string for disk partitions in a Linux system.

**Step 1** Query the partition UUID.

**blkid** *Disk partition*

In this example, the UUID of the `/dev/vdb1` partition is queried.

```
blkid /dev/vdb1
```

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# blkid /dev/vdb1  
/dev/vdb1: UUID="0b3040e2-1367-4abb-841d-ddb0b92693df" TYPE="ext4"
```

Carefully record the UUID, as you will need it for the following step.

**Step 2** Open the `fstab` file using the vi editor.

```
vi /etc/fstab
```

**Step 3** Press `i` to enter editing mode.

**Step 4** Move the cursor to the end of the file and press `Enter`. Then, add the following information:

```
UUID=0b3040e2-1367-4abb-841d-ddb0b92693df /mnt/sdc          ext4  defaults  0 2
```

In this example, the line starting with "UUID" is the information added. Edit this line to match the following format:

- **UUID:** The UUID obtained in [Step 1](#).
- **Mount point:** The directory on which the partition is mounted. You can query the mount point using `df -TH`.
- **Filesystem:** The file system format of the partition. You can query the file system format using `df -TH`.
- **Mount option:** The partition mount option. Usually, this parameter is set to **defaults**.
- **Dump:** The Linux dump backup option.
  - **0:** Linux dump backup is not used. Usually, dump backup is not used, and you can set this parameter to **0**.
  - **1:** Linux dump backup is used.
- **fsck:** The fsck option, which means whether to use fsck to check the disk during startup.
  - **0:** not use fsck.
  - If the mount point is the root partition (`/`), this parameter must be set to **1**.  
If this parameter is set to **1** for the root partition, this parameter for other partitions must start with **2** because the system checks the partitions in the ascending order of the values.

**Step 5** Press `Esc`, enter `:wq`, and press `Enter`.

The system saves the configurations and exits the vi editor.

**Step 6** Verify that the disk is auto-mounted at startup.

1. Unmount the partition.

```
umount Disk partition
```

In this example, run the following command:

```
umount /dev/vdb1
```

2. Reload all the content in the `/etc/fstab` file.

```
mount -a
```

3. Query the file system mounting information.

```
mount | grep Mount point
```

In this example, run the following command:

```
mount | grep /mnt/sdc
```

If information similar to the following is displayed, automatic mounting has been configured:

```
root@ecs-test-0001 ~]# mount | grep /mnt/sdc  
/dev/vdb1 on /mnt/sdc type ext4 (rw,relatime,data=ordered)
```

----End

## 6.3 Initializing a Linux Data Disk (parted)

### Scenarios

This section uses CentOS 7.4 64bit to describe how to initialize a data disk attached to a server running Linux and use parted to partition the data disk.

The maximum partition size that MBR supports is 2 TiB and that GPT supports is 18 EiB. If the disk size you need to partition is greater than 2 TiB, partition the disk using GPT.

The fdisk partitioning tool is suitable only for MBR partitions, and the parted partitioning tool is suitable for both MBR and GPT partitions. For more information, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

The method for initializing a disk varies slightly depending on the OS running on the server. This document is used for reference only. For the detailed operations and differences, see the product documents of the corresponding OS.

---

#### NOTICE

When using a disk for the first time, if you have not initialized it, including creating partitions and file systems, the additional space added to this disk in an expansion later may not be normally used.

---

### Prerequisites

- A data disk has been attached to a server and has not been initialized.
- You have logged in to the server.
  - For how to log in to an ECS, see [Logging In to an ECS](#).
  - For how to log in to a BMS, see [Logging In to the BMS](#).

### Creating and Mounting a Partition

The following example shows you how a new partition can be created on a new data disk that has been attached to a server. The partition will be created using

parted, and GPT will be used. Furthermore, the partition will be formatted using the ext4 file system, mounted on **/mnt/sdc**, and configured to mount automatically at startup.

**Step 1** Query information about the new data disk.

**lsblk**

Information similar to the following is displayed:

```
root@ecs-test-0001 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
vda 253:0 0 40G 0 disk
└─vda1 253:1 0 40G 0 part /
vdb 253:16 0 100G 0 disk
```

In the command output, this server contains two disks. **/dev/vda** and **/dev/vdb**. **/dev/vda** is the system disk, and **/dev/vdb** is the new data disk.

**Step 2** Launch parted to partition the new data disk.

**parted** *New data disk*

In this example, run the following command:

**parted /dev/vdb**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# parted /dev/vdb
GNU Parted 3.1
Using /dev/vdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

**Step 3** Enter **p** and press **Enter** to view the current disk partition style.

Information similar to the following is displayed:

```
(parted) p
Error: /dev/vdb: unrecognised disk label
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GiB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
(parted)
```

In the command output, the **Partition Table** value is **unknown**, indicating that no partition style is set for the new disk.

**Step 4** Set the disk partition style.

**mklabel** *Disk partition style*

This command lets you control whether to use MBR or GPT for your partition table. In this example, GPT is used.

**mklabel gpt**

**NOTICE**

The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

If the partition style is changed after the disk has been used, all data on the disk will be lost, so take care to select an appropriate partition style when initializing the disk. If you must change the partition style to GPT after a disk has been used, it is recommended that you back up the disk data before the change.

**Step 5** Enter **p** and press **Enter** to view the disk partition style.

Information similar to the following is displayed:

```
(parted) mklabel gpt
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 107GiB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number Start End Size File system Name Flags
(parted)
```

In the command output, the **Partition Table** value is **gpt**, indicating that the disk partition style is GPT.

**Step 6** Enter **unit s** and press **Enter** to set the measurement unit of the disk to sector.

**Step 7** Create a new partition.

**mkpart** *Partition name Start sector End sector*

In this example, run the following command:

**mkpart test 2048s 100%**

In this example, one partition is created for the new data disk, starting on **2048** and using **100%** of the rest of the disk. The two values are used for reference only. You can determine the number of partitions and the partition size based on your service requirements.

Information similar to the following is displayed:

```
(parted) mkpart opt 2048s 100%
(parted)
```

**Step 8** Enter **p** and press **Enter** to print the partition details.

Information similar to the following is displayed:

```
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 209715200s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number Start End Size File system Name Flags
1 2048s 209713151s 209711104s test
(parted)
```



**Step 9** Enter **q** and press **Enter** to exit parted.

Information similar to the following is displayed:

```
(parted) q  
Information: You may need to update /etc/fstab.
```

You can configure automatic mounting by updating the **/etc/fstab** file. Before doing so, format the partition with a desired file system and mount the partition on the mount point.

**Step 10** View the disk partition information.

**lsblk**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# lsblk  
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT  
vda 253:0 0 40G 0 disk  
└─vda1 253:1 0 40G 0 part /  
vdb 253:16 0 100G 0 disk  
└─vdb1 253:17 0 100G 0 part
```

In the command output, **/dev/vdb1** is the partition you created.

**Step 11** Format the new partition with a desired file system format.

**mkfs -t** *File system format* **/dev/vdb1**

In this example, the **ext4** format is used for the new partition.

**mkfs -t ext4 /dev/vdb1**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# mkfs -t ext4 /dev/vdb1  
mke2fs 1.42.9 (28-Dec-2013)  
Filesystem label=  
OS type: Linux  
Block size=4096 (log=2)  
Fragment size=4096 (log=2)  
Stride=0 blocks, Stripe width=0 blocks  
6553600 inodes, 26213888 blocks  
1310694 blocks (5.00%) reserved for the super user  
First data block=0  
Maximum filesystem blocks=2174746624  
800 block groups  
32768 blocks per group, 32768 fragments per group  
8192 inodes per group  
Superblock backups stored on blocks:  
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,  
4096000, 7962624, 11239424, 20480000, 23887872  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (32768 blocks): done  
Writing superblocks and filesystem accounting information: done
```

The formatting takes a period of time. Observe the system running status and do not exit.

---

#### NOTICE

The partition sizes supported by file systems vary. Choose an appropriate file system format based on your service requirements.

---

**Step 12** Create a mount point.

```
mkdir Mount point
```

In this example, the `/mnt/sdc` mount point is created.

```
mkdir /mnt/sdc
```

 **NOTE**

The `/mnt` directory exists on all Linux systems. If the mount point cannot be created, it may be that the `/mnt` directory has been accidentally deleted. You can run `mkdir -p /mnt/sdc` to create the mount point.

**Step 13** Mount the new partition on the created mount point.

```
mount Disk partition Mount point
```

In this example, the `/dev/vdb1` partition is mounted on `/mnt/sdc`.

```
mount /dev/vdb1 /mnt/sdc
```

**Step 14** Check the mount result.

```
df -TH
```

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/vda1       ext4      43G   1.9G  39G   5% /
devtmpfs        devtmpfs  2.0G   0   2.0G   0% /dev
tmpfs           tmpfs     2.0G   0   2.0G   0% /dev/shm
tmpfs           tmpfs     2.0G   9.0M  2.0G   1% /run
tmpfs           tmpfs     2.0G   0   2.0G   0% /sys/fs/cgroup
tmpfs           tmpfs     398M   0   398M   0% /run/user/0
/dev/vdb1       ext4     106G   63M  101G   1% /mnt/sdc
```

You should now see that partition `/dev/vdb1` is mounted on `/mnt/sdc`.

 **NOTE**

After the server is restarted, the disk will not be automatically mounted. To configure automount at startup, you will need to modify the `/etc/fstab` file. For details, see [Configuring Automatic Mounting at System Start](#).

----End

## Configuring Automatic Mounting at System Start

The `fstab` file controls what disks are automatically mounted at startup. You can use `fstab` to configure your data disks to mount automatically. This operation will not affect the existing data.

The example here uses UUIDs to identify disks in the `fstab` file. You are advised not to use device names to identify disks in the file because device names are assigned dynamically and may change (for example, from `/dev/vdb1` to `/dev/vdb2`) after a reboot. This can even prevent your disk from booting up.

 **NOTE**

UUID is the unique character string for disk partitions in a Linux system.

**Step 1** Query the partition UUID.

**blkid** *Disk partition*

In this example, the UUID of the `/dev/vdb1` partition is queried.

**blkid /dev/vdb1**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# blkid /dev/vdb1
/dev/vdb1: UUID="0b3040e2-1367-4abb-841d-ddb0b92693df" TYPE="ext4"
```

Carefully record the UUID, as you will need it for the following step.

**Step 2** Open the `fstab` file using the vi editor.

**vi /etc/fstab**

**Step 3** Press `i` to enter editing mode.

**Step 4** Move the cursor to the end of the file and press **Enter**. Then, add the following information:

```
UUID=0b3040e2-1367-4abb-841d-ddb0b92693df /mnt/sdc          ext4    defaults    0 2
```

In this example, the line starting with "UUID" is the information added. Edit this line to match the following format:

- **UUID:** The UUID obtained in [Step 1](#).
- **Mount point:** The directory on which the partition is mounted. You can query the mount point using `df -TH`.
- **Filesystem:** The file system format of the partition. You can query the file system format using `df -TH`.
- **Mount option:** The partition mount option. Usually, this parameter is set to **defaults**.
- **Dump:** The Linux dump backup option.
  - **0:** Linux dump backup is not used. Usually, dump backup is not used, and you can set this parameter to **0**.
  - **1:** Linux dump backup is used.
- **fsck:** The fsck option, which means whether to use fsck to check the disk during startup.
  - **0:** not use fsck.
  - If the mount point is the root partition (`/`), this parameter must be set to **1**.  
If this parameter is set to **1** for the root partition, this parameter for other partitions must start with **2** because the system checks the partitions in the ascending order of the values.

**Step 5** Press **Esc**, enter `:wq`, and press **Enter**.

The system saves the configurations and exits the vi editor.

**Step 6** Verify that the disk is auto-mounted at startup.

1. Unmount the partition.  
**umount** *Disk partition*

In this example, run the following command:

```
umount /dev/vdb1
```

2. Reload all the content in the `/etc/fstab` file.

```
mount -a
```

3. Query the file system mounting information.

```
mount | grep Mount point
```

In this example, run the following command:

```
mount | grep /mnt/sdc
```

If information similar to the following is displayed, automatic mounting has been configured:

```
root@ecs-test-0001 ~]# mount | grep /mnt/sdc  
/dev/vdb1 on /mnt/sdc type ext4 (rw,relatime,data=ordered)
```

----End

## 6.4 Initializing a Linux Data Disk Larger Than 2 TiB (parted)

### Scenarios

This section uses CentOS 7.4 64bit to describe how to use parted to initialize a data disk whose capacity is larger than 2 TiB. In the following operations, the capacity of the example disk is 3 TiB.

The maximum partition size that MBR supports is 2 TiB and that GPT supports is 18 EiB. If the disk size you need to partition is greater than 2 TiB, partition the disk using GPT.

The fdisk partitioning tool is suitable only for MBR partitions, and the parted partitioning tool is suitable for both MBR and GPT partitions. For more information, see [Introduction to Data Disk Initialization Scenarios and Partition Styles](#).

The method for initializing a disk varies slightly depending on the OS running on the server. This document is used for reference only. For the detailed operations and differences, see the product documents of the corresponding OS.

---

#### NOTICE

When using a disk for the first time, if you have not initialized it, including creating partitions and file systems, the additional space added to this disk in an expansion later may not be normally used.

---

### Prerequisites

- A data disk has been attached to a server and has not been initialized.
- You have logged in to the server.
  - For how to log in to an ECS, see [Logging In to an ECS](#).

- For how to log in to a BMS, see [Logging In to the BMS](#).

## Creating and Mounting a Partition

The following example shows you how a new partition can be created on a new data disk that has been attached to a server. The partition will be created using parted, and GPT will be used. Furthermore, the partition will be formatted using the ext4 file system, mounted on `/mnt/sdc`, and configured to mount automatically at startup.

**Step 1** Query information about the new data disk.

### lsblk

Information similar to the following is displayed:

```
[root@ecs-centos74 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
vda 253:0 0 40G 0 disk
├─vda1 253:1 0 1G 0 part /boot
└─vda2 253:2 0 39G 0 part /
vdb 253:16 0 3T 0 disk
```

In the command output, this server contains two disks. `/dev/vda` and `/dev/vdb`. `/dev/vda` is the system disk, and `/dev/vdb` is the new data disk.

**Step 2** Launch parted to partition the new data disk.

### parted *New data disk*

In this example, run the following command:

### parted /dev/vdb

Information similar to the following is displayed:

```
[root@ecs-centos74 ~]# parted /dev/vdb
GNU Parted 3.1
Using /dev/vdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

**Step 3** Enter **p** and press **Enter** to view the current disk partition style.

Information similar to the following is displayed:

```
(parted) p
Error: /dev/vdb: unrecognised disk label
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 3299GiB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
(parted)
```

In the command output, the **Partition Table** value is **unknown**, indicating that no partition style is set for the new disk.

**Step 4** Set the disk partition style.

### mklabel *Disk partition style*

The disk partition style can be MBR or GPT. If the disk capacity is greater than 2 TiB, use GPT.

## mklabel gpt

### NOTICE

The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

If the partition style is changed after the disk has been used, all data on the disk will be lost, so take care to select an appropriate partition style when initializing the disk. If you must change the partition style to GPT after a disk has been used, it is recommended that you back up the disk data before the change.

**Step 5** Enter **p** and press **Enter** to view the disk partition style.

Information similar to the following is displayed:

```
(parted) mklabel gpt
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 3299GiB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start  End  Size  File system  Name  Flags
(parted)
```

**Step 6** Enter **unit s** and press **Enter** to set the measurement unit of the disk to sector.

**Step 7** Create a new partition.

**mkpart** *Partition name Start sector End sector*

In this example, run the following command:

**mkpart opt 2048s 100%**

In this example, one partition is created for the new data disk, starting on **2048** and using **100%** of the rest of the disk. The two values are used for reference only. You can determine the number of partitions and the partition size based on your service requirements.

Information similar to the following is displayed:

```
(parted) mkpart opt 2048s 100%
Warning: The resulting partition is not properly aligned for best performance.
Ignore/Cancel? Ignore
```

If the preceding warning message is displayed, enter **Ignore** to ignore the performance warning.

**Step 8** Enter **p** and press **Enter** to print the partition details.

Information similar to the following is displayed:

```
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 6442450944s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
```

```
Number Start End      Size      File system Name  Flags
1      2048s 6442448895s 6442446848s          opt
```

Details about the **dev/vdb1** partition are displayed.

**Step 9** Enter **q** and press **Enter** to exit parted.

**Step 10** View the disk partition information.

### lsblk

Information similar to the following is displayed:

```
[root@ecs-centos74 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
vda  253:0  0 40G  0 disk
├─vda1 253:1  0  1G  0 part /boot
├─vda2 253:2  0 39G  0 part /
vdb  253:16 0  3T  0 disk
├─vdb1 253:17 0  3T  0 part
```

In the command output, **/dev/vdb1** is the partition you created.

**Step 11** Format the new partition with a desired file system format.

**mkfs -t** *File system format* **/dev/vdb1**

In this example, the **ext4** format is used for the new partition.

**mkfs -t ext4 /dev/vdb1**

Information similar to the following is displayed:

```
[root@ecs-centos74 ~]# mkfs -t ext4 /dev/vdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
201326592 inodes, 805305856 blocks
40265292 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2952790016
24576 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000, 214990848, 512000000, 550731776, 644972544

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

The formatting takes a period of time. Observe the system running status and do not exit.

---

### NOTICE

The partition sizes supported by file systems vary. Choose an appropriate file system format based on your service requirements.

---

**Step 12** Create a mount point.

```
mkdir Mount point
```

In this example, the `/mnt/sdc` mount point is created.

```
mkdir /mnt/sdc
```

 **NOTE**

The `/mnt` directory exists on all Linux systems. If the mount point cannot be created, it may be that the `/mnt` directory has been accidentally deleted. You can run `mkdir -p /mnt/sdc` to create the mount point.

**Step 13** Mount the new partition on the created mount point.

```
mount Disk partition Mount point
```

In this example, the `/dev/vdb1` partition is mounted on `/mnt/sdc`.

```
mount /dev/vdb1 /mnt/sdc
```

**Step 14** Check the mount result.

```
df -TH
```

Information similar to the following is displayed:

```
[root@ecs-centos74 ~]# df -TH
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/vda2       ext4      42G  1.5G  38G   4% /
devtmpfs        devtmpfs  2.0G   0  2.0G   0% /dev
tmpfs           tmpfs     2.0G   0  2.0G   0% /dev/shm
tmpfs           tmpfs     2.0G  8.9M  2.0G   1% /run
tmpfs           tmpfs     2.0G   0  2.0G   0% /sys/fs/cgroup
/dev/vda1       ext4      1.1G  153M  801M  17% /boot
tmpfs           tmpfs     398M   0  398M   0% /run/user/0
/dev/vdb1       ext4      3.3T   93M  3.1T   1% /mnt/sdc
```

You should now see that partition `/dev/vdb1` is mounted on `/mnt/sdc`.

----End

## Configuring Automatic Mounting at System Start

The `fstab` file controls what disks are automatically mounted at startup. You can use `fstab` to configure your data disks to mount automatically. This operation will not affect the existing data.

The example here uses UUIDs to identify disks in the `fstab` file. You are advised not to use device names to identify disks in the file because device names are assigned dynamically and may change (for example, from `/dev/vdb1` to `/dev/vdb2`) after a reboot. This can even prevent your disk from booting up.

 **NOTE**

UUID is the unique character string for disk partitions in a Linux system.

**Step 1** Query the partition UUID.

```
blkid Disk partition
```

In this example, the UUID of the `/dev/vdb1` partition is queried.



**blkid /dev/vdb1**

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# blkid /dev/vdb1
/dev/vdb1: UUID="0b3040e2-1367-4abb-841d-ddb0b92693df" TYPE="ext4"
```

Carefully record the UUID, as you will need it for the following step.

**Step 2** Open the **fstab** file using the vi editor.

**vi /etc/fstab**

**Step 3** Press **i** to enter editing mode.

**Step 4** Move the cursor to the end of the file and press **Enter**. Then, add the following information:

```
UUID=0b3040e2-1367-4abb-841d-ddb0b92693df /mnt/sdc          ext4  defaults  0 2
```

In this example, the line starting with "UUID" is the information added. Edit this line to match the following format:

- **UUID:** The UUID obtained in [Step 1](#).
- **Mount point:** The directory on which the partition is mounted. You can query the mount point using **df -TH**.
- **Filesystem:** The file system format of the partition. You can query the file system format using **df -TH**.
- **Mount option:** The partition mount option. Usually, this parameter is set to **defaults**.
- **Dump:** The Linux dump backup option.
  - **0:** Linux dump backup is not used. Usually, dump backup is not used, and you can set this parameter to **0**.
  - **1:** Linux dump backup is used.
- **fsck:** The fsck option, which means whether to use fsck to check the disk during startup.
  - **0:** not use fsck.
  - If the mount point is the root partition (**/**), this parameter must be set to **1**.

If this parameter is set to **1** for the root partition, this parameter for other partitions must start with **2** because the system checks the partitions in the ascending order of the values.

**Step 5** Press **Esc**, enter **:wq**, and press **Enter**.

The system saves the configurations and exits the vi editor.

**Step 6** Verify that the disk is auto-mounted at startup.

1. Unmount the partition.

**umount** *Disk partition*

In this example, run the following command:

**umount /dev/vdb1**

2. Reload all the content in the **/etc/fstab** file.

**mount -a**

3. Query the file system mounting information.

**mount | grep *Mount point***

In this example, run the following command:

**mount | grep /mnt/sdc**

If information similar to the following is displayed, automatic mounting has been configured:

```
root@ecs-test-0001 ~]# mount | grep /mnt/sdc  
/dev/vdb1 on /mnt/sdc type ext4 (rw,relatime,data=ordered)
```

----End

# A Change History

---

Released On	Description
2023-07-20	This issue is the second official release, which incorporates the following change: Updated and added constraints in sections <a href="#">Step 2: Purchase an EVS Disk</a> , <a href="#">Attaching a Non-Shared Disk</a> , <a href="#">Attaching a Shared Disk</a> , and <a href="#">Introduction to Data Disk Initialization Scenarios and Partition Styles</a> .
2022-09-30	This issue is the first official release.