Elastic Volume Service

User Guide

Issue 01

Date 2022-08-08

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Contents

1 Overview

1.1 What Is EVS?

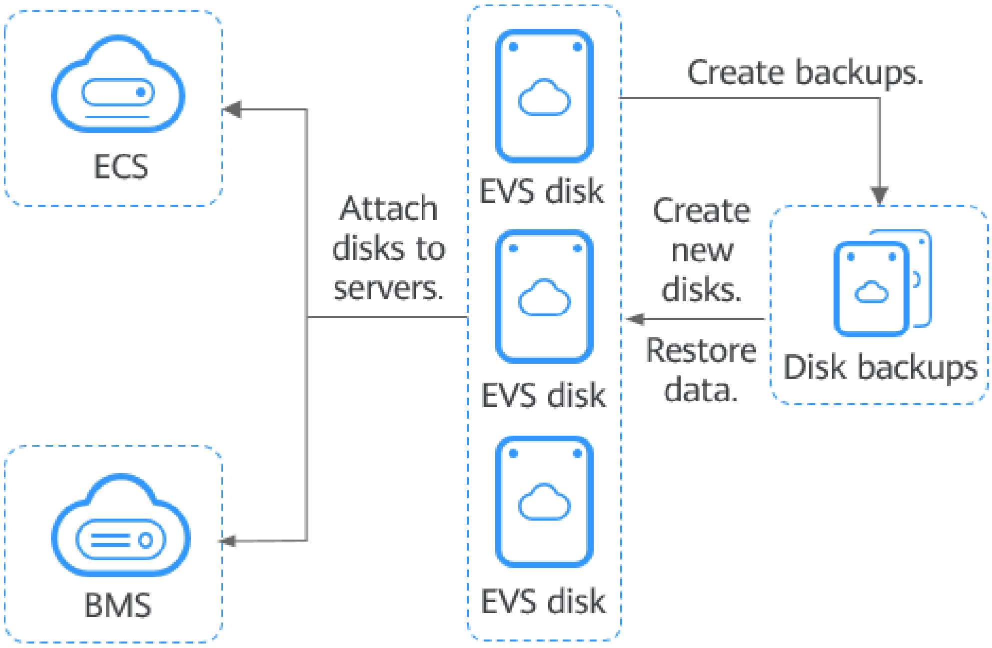
Overview

Elastic Volume Service (EVS) offers scalable block storage for cloud servers. With high reliability, high performance, and a variety of specifications, EVS disks can be used for distributed file systems, development and test environments, data warehouses, and high-performance computing (HPC) applications. Cloud servers that EVS supports include Elastic Cloud Servers (ECSs) and Bare Metal Servers (BMSs).

EVS disks are similar to hard disks in PCs. They must be attached to servers for use and cannot be used alone. You can initialize EVS disks, create file systems on them, and store data persistently on them.

EVS disks are sometimes just referred to as disks in this document.

Figure 1-1 EVS architecture



EVS Advantages

EVS has the following advantages:

Table 1-1 EVS advantages

Advantage	Description	Related Knowledge
Various disk types	EVS provides a variety of disk types for you to choose from, and EVS disks can be used as data disks and system disks for servers. You can select an appropriate disk type that best suits your budget and service requirements.	**Disk Types and Performance**
Elastic scalability	The EVS disk capacity ranges from 10 GiB to 32 TiB. When it no longer meets your needs, you can expand the disk capacity up to 32 TiB in increments of 1 GiB, without interrupting your applications.	**Expansion Overview**
	Besides the disk capacity limit, the additional space you can add cannot exceed the remaining quota. You can increase the quota if the remaining quota is insufficient.	**Managing Quotas**
High security and reliability	Both system disks and data disks support data encryption to ensure data security.	**EVS Encryption**

Advantage	Description	Related Knowledge
	Data protection functions, such as backups, safeguard the disk data, preventing incorrect data caused by application exceptions or attacks.	**EVS Backup**
Real-time monitoring	On Cloud Eye, you can monitor the disk health and operating status at any time.	**Viewing EVS Monitoring Data**

Differences Among EVS, SFS, and OBS

There are currently three types of storage available for you to choose from: EVS, Scalable File Service (SFS), and Object Storage Service (OBS). See their differences in the following table.

Table 1-2 Comparison among SFS, OBS, and EVS

Dimension	SFS	OBS	EVS
Concept	SFS provides on-demand high-performance file storage, which can be shared by multiple servers. SFS is similar to a remote directory for Windows or Linux OSs.	OBS provides massive, secure, reliable, and cost-effective data storage for users to store data of any type and size.	EVS provides scalable block storage that features high reliability and high performance to meet a variety of service requirements. An EVS disk is similar to a hard disk on a PC.
Data storage logic	Stores files. Data is sorted and displayed in files and folders.	Stores objects. Files can be stored directly to OBS. The files automatically generate corresponding system metadata. You can also customize the metadata if needed.	Stores binary data and cannot directly store files. To store files, you need to format the file system first.

Dimension	SFS	OBS	EVS
Access method	SFS file systems can be accessed only after being mounted to ECSs or BMSs through NFS or CIFS. A network address must be specified or mapped to a local directory for access.	OBS buckets can be accessed through the Internet or Direct Connect. The bucket address must be specified for access, and transmission protocols HTTP and HTTPS are used.	EVS disks can be used and accessed from applications only after being attached to ECSs or BMSs and initialized.
Application Scenarios	Gene sequencing, image rendering, media processing, file sharing, content management, and web services	Big data analytics, static website hosting, online video on demand (VoD), gene sequencing, and intelligent video surveillance	Industrial design, energy exploration, critical clustered applications, enterprise application systems, and development and testing
Capacity	Petabytes	Exabytes	Terabytes
Latency	3–10 ms	10 ms	1 ms
IOPS/TPS	10,000 for a single file system	Tens of millions	33,000 for a single disk
Bandwidth	GiB/s	TiB/s	MiB/s
Data sharing	Supported	Supported	Supported
Remote access	Supported	Supported	Not supported
Used independently	Supported	Supported	Not supported

Methods of Access

The public cloud system provides a web-based management console and HTTPS-based APIs for you to access the EVS service.

- APIs

  Use APIs if you need to integrate EVS into a third-party system for secondary development. For details, see *Elastic Volume Service API Reference*.
- Management console

Use the management console if you do not need to integrate EVS with a third-party system. Log in to the management console and choose **Elastic Volume Service** from the service list.

User Permissions

When EVS is used, the public cloud system provides two types of user permissions by default: user management and resource management.

- User management refers to the management of users, user groups, and user group rights.
- Users with resource management permissions can control the operations performed on cloud service resources.

1.2 Disk Types and Performance

EVS disks are classified based on the disk I/O performance. EVS disks differ in performance and price. Choose the disk type most appropriate for your applications.

Application Scenarios

- Common I/O: This type of EVS disks delivers a maximum IOPS of 1,000. They are suitable for applications that require large capacity, a medium read/write speed, and fewer transactions, such as enterprise office applications and small-scale test environments.
- Ultra-high I/O: This type of EVS disks delivers a maximum IOPS of 20,000 and a minimum read/write latency of 1 ms. They are excellent for read/write-intensive applications that require super-high I/O and bandwidth, such as distributed file systems in HPC scenarios or NoSQL/relational databases in I/O-intensive scenarios.
- Extreme SSD: This type of EVS disks delivers a maximum IOPS of 128,000. They are designed for workloads demanding super-high bandwidth and super-low latency, such as Oracle databases and AI applications.
- General Purpose SSD: This type of EVS disks delivers a maximum IOPS of 20,000. They are suitable for workloads requiring high throughput and low latency, such as enterprise OA, development and testing, web server logging, containers, and high-performance system disks.

NOTE

If an Extreme SSD disk is attached to a BMS, it can reach a maximum IOPS of 128,000. If it is attached to an ECS, it can reach a maximum IOPS of 100,000 due to I/O queue limitations.

EVS Performance

EVS performance metrics include:

- IOPS: Number of read/write operations performed by an EVS disk per second
- Throughput: Amount of data read from and written into an EVS disk per second

- Read/write I/O latency: Minimum interval between two consecutive read/write operations on an EVS disk

  Single-queue access latencies of different types of EVS disks are as follows:
  - General Purpose SSD: 1 ms
  - Common I/O: 10 ms to 15 ms
  - Ultra-high I/O: 1 ms to 3 ms
  - Extreme SSD: sub-millisecond level

Table 1-3 EVS performance data

Parameter	Common I/O	Ultra-high I/O
IOPS per GiB/EVS disk	1	50
Min. IOPS/EVS disk	100	1,500
Max. IOPS/EVS disk	1,000	20,000
IOPS burst limit/EVS disk	1,000	10,000
Max. throughput	40 MiB/s	320 MiB/s

NOTE

Performance data listed in **Table 1-3** is applicable to the EU-Paris region only.

Calculating Disk IOPS Limit

To calculate the IOPS limit of a disk, you need to obtain the smaller value of the following two values:

- Max. IOPS/EVS disk
- Baseline IOPS/EVS disk + IOPS per GiB x Disk capacity

The following example uses an ultra-high I/O EVS disk with a maximum IOPS of 50,000.

- If the disk capacity is 100 GiB, the disk IOPS limit is calculated as follows:

  Disk IOPS limit = Min. (50,000, 1,800 + 50 x 100)

  The disk IOPS limit is 6,800, the smaller value between 50,000 and 6,800.
- If the disk capacity is 1,000 GiB, the disk IOPS limit is calculated as follows:

  Disk IOPS limit = Min. (50,000, 1,800 + 50 x 1,000)

  The disk IOPS limit is 50,000, the smaller value between 50,000 and 51,800.

Disk IOPS Calculation Formula

Disk IOPS = Min. (Maximum IOPS, Minimum IOPS + IOPS per GiB x Capacity)

The following example uses an ultra-high I/O EVS disk with a maximum IOPS of 50,000.

- If the disk capacity is 100 GiB, the disk IOPS is calculated as follows: Disk IOPS = Min. (50,000, 1,800 + 50 × 100)

  The disk IOPS is 6,800, the smaller value between 50,000 and 6,800.
- If the disk capacity is 1,000 GiB, the disk IOPS is calculated as follows: Disk IOPS = Min. (50,000, 1,800 + 50 × 1,000)

  The disk IOPS is 50,000, the smaller value between 50,000 and 51,800.

Disk Burst Capability and Principles

EVS disks have burst capability, which allows a small-capacity disk to surpass its maximum IOPS within a certain period of time. This IOPS applies to individual disks.

Disks with burst capability are well-suited for speeding up server startup. In most cases, system disks have small capacities. For example, the IOPS of a 50-GiB ultra-high I/O disk without burst capability can only reach up to 4,300, calculated as follows: IOPS = Min. (50,000, 1,800 + 50 x Capacity). If the disk has burst capability, its IOPS can burst up to 16,000.

The following example uses an ultra-high I/O EVS disk with the IOPS burst limit of 16,000.

- If the disk capacity is 100 GiB, the disk has a maximum IOPS of 6,800, but it can burst to 16,000 IOPS in a certain duration.
- If the disk capacity is 1,000 GiB, the disk has a maximum IOPS of 50,000. The disk maximum IOPS already exceeds its burst IOPS 16,000, and the disk does not use the burst capability.

The following describes the burst IOPS consumption and reservation.

A token bucket is used to handle burst I/O operations. The number of initial tokens in the bucket is calculated as follows:

Number of initial tokens = Burst duration x IOPS burst limit

In the following example, a 100-GiB ultra-high I/O EVS disk is used, and the fixed burst duration is 1800s. Therefore, the number of initial tokens is 28,800,000 (1,800 x 16,000).

- Token production rate: This rate equals the disk maximum IOPS, which is 6,800 tokens/s.
- Token consumption rate: This rate is calculated based on the I/O usage. Each I/O request consumes a token. The maximum consumption rate is 16,000 tokens/s, which is the larger value between the disk burst IOPS and maximum IOPS.

Consumption principles

When the token consumption rate is greater than the production rate, the number of tokens decreases accordingly, and eventually the disk IOPS will be consistent with the token production rate (the maximum IOPS). In this example, the disk can burst for approximately 3,130 seconds [28,800,000/(16,000 - 6,800)].

Reservation principles

When the token consumption rate is smaller than the production rate, the number of tokens increases accordingly, enabling the disk to regain the burst capability. In this example, if the disk is suspended for approximately 4,235 seconds (28,800,000/6,800), the token bucket will be filled up with tokens.

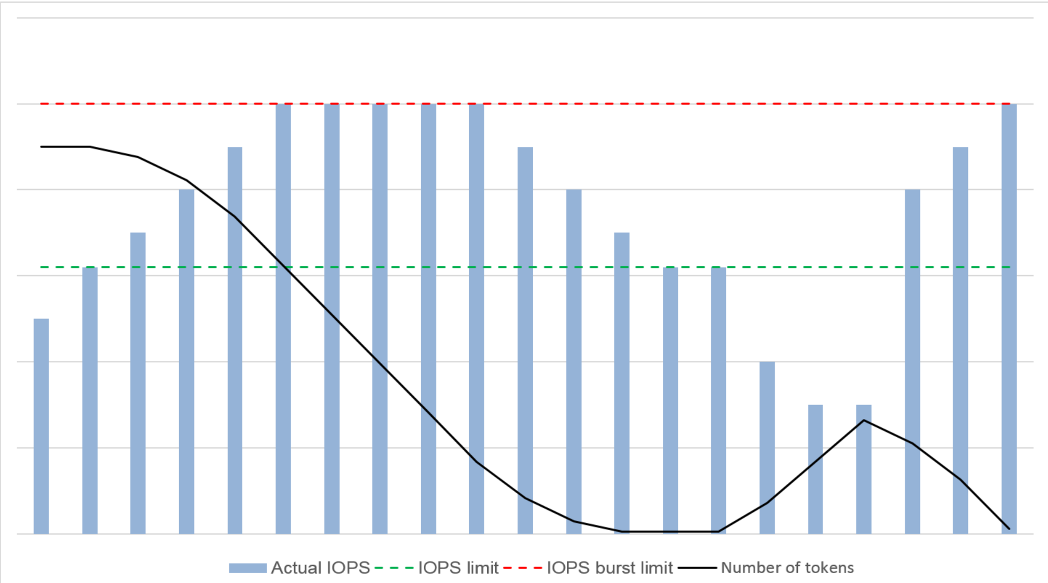
NOTE

As long as there are tokens in the token bucket, the disk has the burst capability.

Figure 1-2 shows the token consumption and reservation principles. The blue bars indicate the disk IOPS usage, the green dashed line represents the maximum IOPS, the red dashed line indicates the IOPS burst limit, and the black curve indicates the changes of the number of tokens.

- When the number of tokens is greater than zero, the disk IOPS can exceed 6,800 and has the capability to reach 16,000, the IOPS burst limit.
- When the number of tokens is zero, the disk does not have the burst capability, and the maximum IOPS is 6,800.
- When the disk IOPS is less than 6,800, the number of tokens starts to increase, and the disk can regain the burst capability.

Figure 1-2 Burst capability diagram



Performance Test Method

For details about how to test the EVS disk performance parameters, see **How Can I Test My Disk Performance?**.

1.3 Device Types and Usage Instructions

What Device Types Are Available?

There are two EVS device types: Virtual Block Device (VBD) and Small Computer System Interface (SCSI).

- VBD is the default EVS device type. VBD EVS disks support only basic read/write SCSI commands.

- SCSI EVS disks support transparent SCSI command transmission and allow the server OS to directly access the underlying storage media. Besides basic read/write SCSI commands, SCSI disks support advanced SCSI commands.

Device type is configured during creation. It cannot be changed after the disk has been created.

Common Application Scenarios and Usage Instructions of SCSI EVS Disks

- BMSs support only SCSI EVS disks.
- Shared SCSI EVS disks: Shared SCSI EVS disks must be used together with a distributed file system or cluster software. Because most cluster applications, such as Windows MSCS, Veritas VCS, and Veritas CFS, require SCSI reservations, you are advised to use shared EVS disks with SCSI.

  SCSI reservations take effect only when shared SCSI EVS disks are attached to ECSs in the same ECS group. For more information about shared EVS disks, see **Shared EVS Disks and Usage Instructions**.

Do I Need to Install a Driver for SCSI EVS Disks?

To use SCSI EVS disks, a cloud server must have a SCSI driver installed. If the SCSI driver is not pre-installed, you need to install it manually.

Check whether you need to manually install the driver based on the server type.

- Bare Metal Server (BMS)

  Both the Windows and Linux images for BMSs are pre-installed with the required SDI card driver. Therefore, no driver needs to be installed.
- KVM ECS

  You are advised to use SCSI EVS disks with KVM ECSs. Linux images and Windows images for KVM ECSs already have the required driver. Therefore, no driver needs to be installed for KVM ECSs.

  NOTE

  ECS virtualization types are categorized into KVM and Xen. For details, see **Product Introduction** > **ECS Types** in the *Elastic Cloud Server User Guide*.
- Xen ECS

  Due to driver limitations, you are advised not to use SCSI EVS disk with Xen ECSs.

  However, a few images support SCSI EVS disks on Xen ECSs. For the supported images, see **Table 1-4**.

NOTE

After confirming that the OS images of Xen ECSs support SCSI EVS disks, determine whether you need to install the driver:

- Public Windows images are preinstalled with the Paravirtual SCSI (PVSCSI) driver. Therefore, no driver needs to be installed.
- Private Windows images are not preinstalled with the PVSCSI driver. You need to download and install it explicitly.

  For details, see **(Optional) Optimizing Windows Private Images** in the *Image Management Service User Guide*.
- Linux images are not preinstalled with the PVSCSI driver. You need to obtain the source code of the open-source Linux driver at **https://github.com/UVP-Tools/SAP-HANA-Tools**.

Table 1-4 OSs supporting SCSI EVS disks

Virtualization Type	OS	
Xen	Windows	See the Windows images listed on the **Public Images** page. Log in to the management console, choose **Image Management Service**, click the **Public Images** tab, and select **ECS image** and **Windows** from the drop-down lists, respectively.
	Linux	• SUSE Linux Enterprise Server 11 SP4 64bit (The kernel version is 3.0.101-68-default or 3.0.101-80-default.) • SUSE Linux Enterprise Server 12 64bit (The kernel version is 3.12.51-52.31-default.) • SUSE Linux Enterprise Server 12 SP1 64bit (The kernel version is 3.12.67-60.64.24-default.) • SUSE Linux Enterprise Server 12 SP2 64bit (The kernel version is 4.4.74-92.35.1-default.)

1.4 Shared EVS Disks and Usage Instructions

What Are Shared EVS Disks?

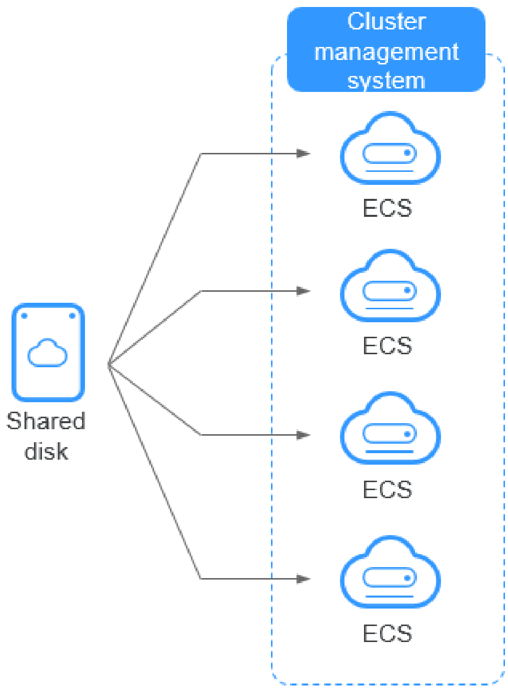
Shared EVS disks are block storage devices that support concurrent read/write operations and can be attached to multiple servers. Shared EVS disks feature multiple attachments, high-concurrency, high-performance, and high-reliability. They are usually used for enterprise business-critical applications that require cluster deployment. Multiple servers can access the same shared EVS disk at the same time.

A shared EVS disk can be attached to a maximum of 16 servers. Servers that EVS supports include ECSs and BMSs. To share files, you need to deploy a shared file system or a cluster management system, such as Windows MSCS, Veritas VCS, or CFS.

NOTICE

You must set up a shared file system or cluster management system before using shared EVS disks. If you directly attach a disk to multiple servers, the sharing function will not work and data may be overwritten.

Figure 1-3 Application scenario of shared EVS disks



Usage Precautions

Because most cluster applications, such as Windows MSCS, Veritas VCS, and Veritas CFS, require SCSI reservations, you are advised to use shared EVS disks with SCSI. If a SCSI EVS disk is attached to a Xen ECS for use, you must install the driver. For details, see **Device Types and Usage Instructions**.

You can create shared VBD disks or shared SCSI disks. It is recommended that you attach a shared disk to the ECSs in the same ECS group to improve service reliability.

- Shared VBD disks: The device type of a newly created shared disk is VBD by default. Such disks can be used as virtual block storage devices, but do not support SCSI reservations. If SCSI reservations are required for your applications, create shared SCSI EVS disks.
- Shared SCSI disks: Such disks support SCSI reservations.

NOTICE

- To improve data security, you are advised to use SCSI reservations together with the anti-affinity policy of an ECS group. That said, ensure that shared SCSI disks are only attached to ECSs in the same anti-affinity ECS group.
- If an ECS does not belong to any anti-affinity ECS group, you are advised not to attach shared SCSI disks to this ECS. Otherwise, SCSI reservations may not work properly, which may put your data at risk.

Concepts of the anti-affinity ECS group and SCSI reservations:

– The anti-affinity policy of an ECS group allows ECSs to be created on different physical servers to improve service reliability.

  For details about ECS groups, see **Managing ECS Groups** in the *Elastic Cloud Server User Guide*.

– The SCSI reservation mechanism uses a SCSI reservation command to perform SCSI reservation operations. If an ECS sends such a command to an EVS disk, the disk is displayed as locked to other ECSs, preventing the data damage that may be caused by simultaneous read/write operations to the disk from multiple ECSs.

– ECS groups and SCSI reservations have the following relationship: A SCSI reservation on a single EVS disk cannot differentiate multiple ECSs on the same physical host. For that reason, if multiple ECSs that use the same shared EVS disk are running on the same physical host, SCSI reservations will not work properly. So you are advised to use SCSI reservations only on ECSs that are in the same ECS group, thus having a working anti-affinity policy.

Advantages

- Multiple attachments: A shared EVS disk can be attached to a maximum of 16 servers.
- High-performance: The random read/write IOPS of a shared ultra-high I/O disk can reach up to 160,000.
- High-reliability: Shared EVS disks support both manual and automatic backup, delivering highly reliable data storage.
- Wide range of use: Shared EVS disks can be used for Linux RHCS clusters where only VBD EVS disks are needed. They can also be used for Windows MSCS and Veritas VCS clusters that require SCSI reservations.

Specifications and Performance

Shared EVS disks have the same specifications and performance as non-shared EVS disks.

Data Sharing Principle and Common Usage Mistakes

A shared EVS disk is essentially the disk that can be attached to multiple servers for use, which is similar to a physical disk in that the disk can be attached to multiple physical servers, and each server can read data from and write data into any space on the disk. If the data read/write rules, such as the read/write sequence and meaning, between these servers are not defined, data read/write interference between servers or other unpredictable errors may occur.

Though shared EVS disks are block storage devices that provide shared access for servers, shared EVS disks do not have the cluster management capability. Therefore, you need to deploy a cluster system to manage shared EVS disks. Common cluster management systems include Windows MSCS, Linux RHCS, Veritas VCS, and Veritas CFS.

If shared EVS disks are not managed by a cluster system, the following issues may occur:

- Data inconsistency caused by read/write conflicts

  When a shared EVS disk is attached to two servers (server A and server B), server A cannot recognize the disk spaces allocated to server B, vice versa. That said, a disk space allocated to server A may be already used by server B. In this case, repeated disk space allocation occurs, which leads to data errors.

  For example, a shared EVS disk has been formatted into the ext3 file system and attached to server A and server B. Server A has written metadata into the file system in space R and space G. Then server B has written metadata into space E and space G. In this case, the data written into space G by server A will be replaced. When the metadata in space G is read, an error will occur.

- Data inconsistency caused by data caching

  When a shared EVS disk is attached to two servers (server A and server B), the application on server A has read the data in space R and space G, then cached the data. At that time, other processes and threads on server A would then read this data directly from the cache. At the same time, if the application on server B has modified the data in space R and space G, the application on server A cannot detect this data change and still reads this data from the cache. As a result, the user cannot view the modified data on server A.

  For example, a shared EVS disk has been formatted into the ext3 file system and attached to server A and server B. Both servers have cached the metadata in the file system. Then server A has created a new file (file F) on the shared disk, but server B cannot detect this modification and still reads data from its cached data. As a result, the user cannot view file F on server B.

Before you attach a shared EVS disk to multiple servers, the disk device type needs to be determined. The device type can be either VBD or SCSI. Shared SCSI EVS disks support SCSI reservations. Before using SCSI reservations, you need to install a driver in the server OS and ensure that the OS image is included in the compatibility list.

For details about the usages of shared EVS disks, see **Managing Shared EVS Disks**.

NOTICE

If you simply attach a shared EVS disk to multiple servers, files cannot be shared between the servers as shared EVS disks do not have the cluster capability. Therefore, build a shared file system or deploy a cluster management system if you need to share files between servers.

1.5 EVS Encryption

What Is EVS Encryption?

In case your services require encryption for the data stored on EVS disks, EVS provides you with the encryption function. You can encrypt newly created EVS disks. EVS uses the industry-standard XTS-AES-256 encryption algorithm and keys to encrypt EVS disks. Keys used by encrypted EVS disks are provided by the Key Management Service (KMS), which is secure and convenient. So you do not need to establish and maintain the key management infrastructure. KMS uses the Hardware Security Module (HSM) that complies with FIPS 140-2 level 3 requirements to protect keys. All user keys are protected by the root key in HSM to prevent key exposure.

NOTICE

The encryption attribute of a disk cannot be changed after the disk is created.

Keys Used for EVS Encryption

Keys provided by KMS include a Default Master Key and Customer Master Keys (CMKs).

- Default Master Key: A key that is automatically created by EVS through KMS and named **evs/default**.

  It cannot be disabled and does not support scheduled deletion.

- CMKs: Keys created by users. You may use existing CMKs or create new CMKs to encrypt disks. For details, see **Management** > **Creating a CMK** in the *Key Management Service User Guide*.

When an encrypted disk is attached, EVS accesses KMS, and KMS sends the data key (DK) to the host memory for use. The disk uses the DK plaintext to encrypt and decrypt disk I/Os. The DK plaintext is only stored in the memory of the host housing the ECS and is not stored persistently on the media. If a CMK is disabled or deleted in KMS, the disk encrypted using this CMK can still use the DK plaintext stored in the host memory. If this disk is later detached, the DK plaintext will be deleted from the memory, and data can no longer be read from or written to the disk. Before you re-attach this encrypted disk, ensure that the CMK is enabled.

If you use a CMK to encrypt disks and this CMK is then disabled or scheduled for deletion, data cannot be read from or written to these disks or may never be restored. See **Table 1-5** for more information.

Table 1-5 Impact of CMK unavailability

<table>
<tr><th>CMK Status</th><th>Impact</th><th>How to Restore</th></tr>
<tr><td>Disabled</td><td rowspan="3">For an encrypted disk already attached:
Reads and writes to the disk are normal unless the disk is detached. Once detached, the disk cannot be attached again.
For an encrypted disk not attached:
The disk cannot be attached anymore.</td><td>Enable the CMK. For details, see Managing CMKs > Enabling One or More CMKs in the Key Management Service User Guide.</td></tr>
<tr><td>Scheduled deletion</td><td>Cancel the scheduled deletion for the CMK. For details, see Managing CMKs > Canceling the Scheduled Deletion of One or More CMKs in the Key Management Service User Guide.</td></tr>
<tr><td>Deleted</td><td>Data on the disks can never be restored.</td></tr>
</table>

Encryption Scenarios

- **System disk encryption**

  System disks are created along with servers and cannot be created separately. So whether a system disk is encrypted or not depends on the image selected during the server creation. See the following table for details.

  Table 1-6 Encryption relationship between images and system disks

Creating Server Using Encrypted Image	Whether System Disk Will Be Encrypted	Description
Yes	Yes	For details, see **Managing Private Images** > **Encrypting Images** in the *Image Management Service User Guide*.
No	No	-

- **Data disk encryption**

  Data disks can be created along with servers or separately. Whether data disks are encrypted depends on their data sources. See the following table for details.

Table 1-7 Encryption relationship between backups, images, and data disks

Created On	Method of Creation	Whether Data Disk Will Be Encrypted	Description
The ECS console	Created together with the server	Yes/No	When a data disk is created together with a server, you can choose to encrypt the disk or not. For details, see **Getting Started** > **Creating an ECS** > **Step 1: Configure Basic Settings** in the *Elastic Cloud Server User Guide*.
The EVS console	No data source selected	Yes/No	When an empty disk is created, you can choose whether to encrypt the disk or not. The encryption attribute of the disk cannot be changed after the disk has been created.
	Creating from a backup	Yes/No	● When a disk is created from a backup, you can choose whether to encrypt the disk or not. The encryption attributes of the disk and backup do not need to be the same. ● When you create a backup for a system or data disk, the encryption attribute of the backup will be the same as that of the disk.
	Creating from an image (The image's source disk is encrypted.)	Yes	-
	Creating from an image (The image's source disk is not encrypted.)	No	-

Who Can Use the Encryption Function?

When a user uses the encryption function, the condition varies depending on whether the user is the first one ever in the current region or project to use this function.

- If the user is the first user, the user needs to follow the prompt to create an agency, which grants KMS Administrator permissions to EVS. Then the user can create and obtain keys to encrypt and decrypt disks.

  NOTE

  The first user must have the KMS Administrator permissions to create the agency. If the user does not have the KMS Administrator permissions, contact the account administrator to grant the permissions first.

- If the user is not the first user, the user can use encryption directly.

1.6 EVS Backup

What Is EVS Backup?

You can back up common I/O, high I/O, and ultra-high I/O disks.

Volume Backup Service (VBS) allows you to create backups for EVS disks while servers are running. If data loss or damage occurred due to virus invasions, accidental deletions, or software/hardware faults, you can use backups to restore data, guaranteeing your data integrity and security.

For more information, see the *Cloud Backup and Recovery User Guide*.

For more information, see the *Volume Backup Service User Guide*.

CBR Architecture

CBR involves backups, vaults, and policies.

Backup

A backup is a copy of a particular chunk of data and is usually stored elsewhere so that it may be used to restore the original data in the event of data loss. There are the following types of backups:

- Cloud disk backup: provides snapshot-based backups for EVS disks.
- Cloud server backup: uses the consistency snapshot technology to protect data for ECSs. Backups of non-database servers are non-database server backups, and those of database servers are application-consistent backups.
- SFS Turbo backup: backs up data of SFS Turbo file systems.

Vault

CBR stores backups in vaults. Before creating a backup, you need to create at least one vault and associate the resources you want to back up with the vaults. Then the resources can be backed up to the associated vaults.

Vaults can be either backup vaults or replication vaults. Backup vaults store resource backups, and replication vaults store backup replicas.

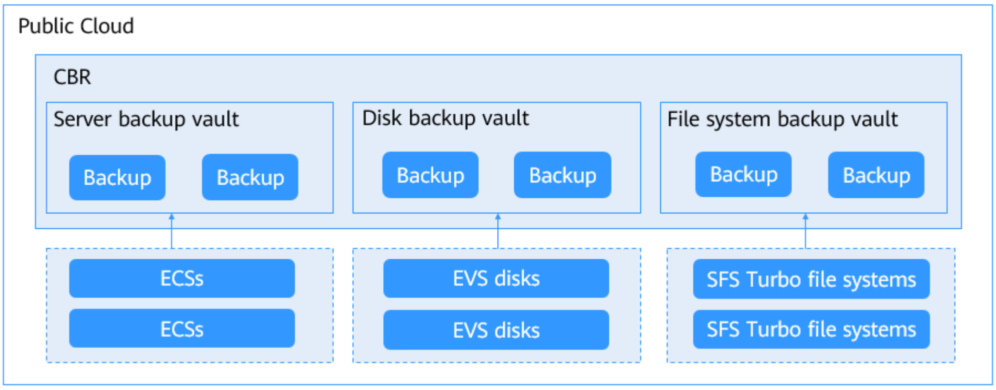
Different types of resources must be backed up to different types of vaults. For example, cloud servers must be backed up to server backup vaults, not disk backup vaults or any other types of vaults.

Policy

There are backup policies and replication policies.

- A backup policy defines when you want to take a backup and for how long you would retain each backup.
- A replication policy defines when you want to replicate from backup vaults and for how long you would retain each replica. Backup replicas are stored in replication vaults.

Figure 1-4 CBR architecture



Application Scenarios

Create and apply backup policies to schedule periodic backups for your EVS disks. You can use the backup data to create new EVS disks or restore to source disks.

Usage Instructions

For how to back up EVS disks, see **Managing EVS Backups** or *Cloud Backup and Recovery User Guide*.

1.7 EVS Three-Copy Redundancy

What Is the Three-Copy Redundancy?

The backend storage system of EVS employs three-copy redundancy to guarantee data reliability. With this mechanism, one piece of data is by default divided into multiple 1 MiB data blocks. Each data block is saved in three copies, and these copies are stored on different nodes in the system according to the distributed algorithms.

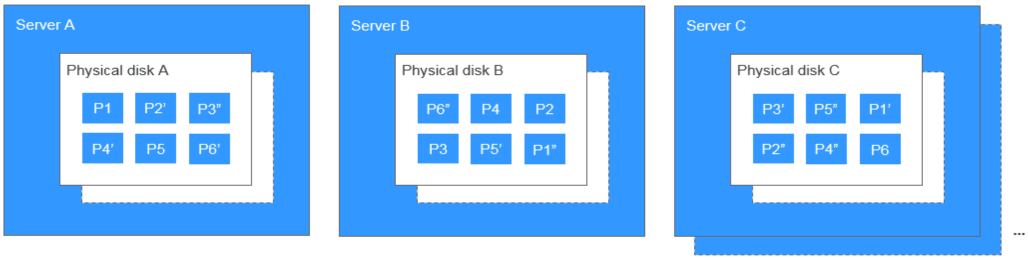
Three-copy redundancy has the following characteristics:

- The storage system saves the data copies on different disks of different servers across cabinets, ensuring that services are not interrupted if a physical device fails.

- The storage system guarantees strong consistency between the data copies.

For example, for data block P1 on physical disk A of server A, the storage system backs up its data to P1'' on physical disk B of server B and to P1' on physical disk C of server C. Data blocks P1, P1', and P1'' are the three copies of the same data block. If physical disk A where P1 resides is faulty, P1' and P1'' can continue providing storage services, ensuring service continuity.

Figure 1-5 Three-copy redundancy



How Does the Three-Copy Redundancy Keep Data Consistency?

When an application writes a piece of data to the system, the three copies of the data in the storage system must be consistent. When any of the three copies is read by the application later, the data on this copy is consistent with the data previously written to it.

Three-copy redundancy keeps data consistency in the following ways:

- Data is simultaneously written to the three copies of the data.

  When an application writes data, the storage system writes it to the three copies of the data simultaneously. In addition, the system returns the write success response to the application only after the data has been written to all of the three copies.
- Storage system automatically restores the damaged copy in the event of a data read failure.

  When an application fails to read data, the system automatically identifies the failure cause. If the data cannot be read from a physical disk sector, the system reads the data from another copy of the data on another node and writes it back to the original disk sector. This ensures the correct number of data copies and data consistency among data copies.

How Does Three-Copy Redundancy Rapidly Rebuild Data?

Each physical disk in the storage system stores multiple data blocks, whose copies are scattered on the nodes in the system according to certain distribution rules. When a physical server or disk fault is detected, the storage system automatically rebuilds the data. Since the copies of data blocks are scattered on different nodes, the storage system will start the data rebuild on multiple nodes simultaneously during a data restore, with only a small amount of data on each node. In this way, the system eliminates the potential performance bottlenecks that may occur when a large amount of data needs to be rebuilt on a single node, and therefore minimizes the adverse impacts exerted on upper-layer applications.

Figure 1-6 shows the data rebuild process.

Figure 1-6 Data rebuild process

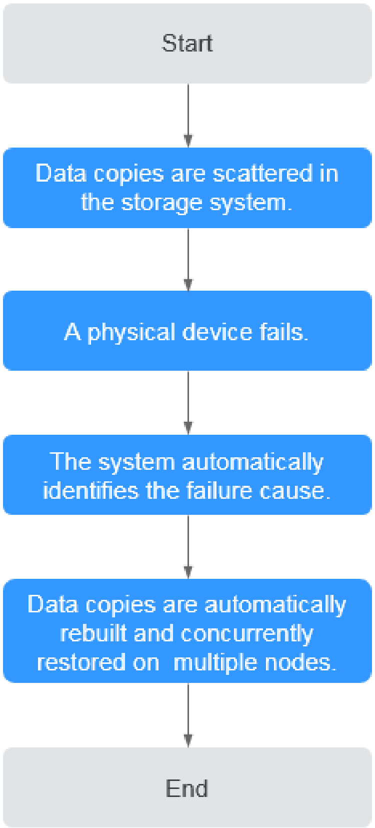


Figure 1-7 shows the data rebuild principle. For example, if physical disks on server F are faulty, the data blocks on these physical disks will be rebuilt on the physical disks of other servers.

Figure 1-7 Data rebuild principle

What Are the Differences Between Three-Copy Redundancy and EVS Backups?

Three-copy redundancy improves the reliability of the data stored on EVS disks. It is used to tackle data loss or inconsistency caused by physical device faults.

EVS backups are used to prevent data loss or inconsistency caused by incorrect operations, viruses, or hacker attacks. So you are advised to create backups to back up the disk data on a timely basis.

1.8 Billing

1.8.1 Billing for Disks

Billing Items

EVS disks are billed based on the disk type, size, and usage duration.

- Billing starts: You will be billed for the EVS disks right after you have purchased them, regardless of whether they are attached or not.
- Billing stops: The billing of a pay-per-use disk stops after it is successfully deleted.

Billing Modes

EVS disks are billed by disk capacity on a pay-per-use basis.

Pay-per-use is a postpaid payment method. Although EVS disks are billed by the second, billing is calculated hourly. If the usage is less than an hour, you are billed based on the actual time consumed.

- Yearly/Monthly is a prepaid payment method.
- Pay-per-use is a postpaid payment method. Although EVS disks are billed by the second, billing is calculated hourly. If the usage is less than an hour, you are billed based on the actual time consumed.

Billing Involved in Configuration Modifications

Item	Yearly/Monthly	Pay-per-Use
Capacity change	• EVS does not support the reduction of disk capacities. • EVS supports the expansion of disk capacities. Additional capacities need to be paid. **NOTE** The expiration time of the EVS disk remains unchanged after the capacity expansion.	• EVS does not support the reduction of disk capacities. • EVS supports the expansion of disk capacities. Multiple pieces of billing records will be generated within a billing cycle (an hour) when an expansion succeeded. For example, if you expand the capacity of an EVS disk from 100 GiB to 200 GiB at 01:30:01, two billing records will be generated in the billing cycle from 01:00:00 to 02:00:00. One is the billing record generated for the 100 GiB from 01:00:00-01:30:00, and the other is the billing record generated for the 200 GiB from 01:30:01-02:00:00.

Renewal

You can renew a yearly/monthly disk anytime before it expires or during the retention period.

You can renew your resources on the Renewals page of the management console. For details, see section "Renewal Management" in the Billing Center User Guide.

Overdue Payment

If your account is not topped up after the account balance falls below zero, your account is in arrears and your pay-per-use disk will enter the retention period. For details, see **Pay-per-Use Disk Usage Suggestions**.

- During the retention period, if you top up your account, the disk will be unfrozen.
- During the retention period, if you do not top up your account, the disk will be released after the retention period ends.

1.8.2 Pay-per-Use Disk Usage Suggestions

Usage Suggestions on Pay-per-Use Resources

If you no longer need to use a pay-per-use disk, you can log in to the management console, detach the disk, and then delete it. For how to delete a disk, see **Deleting EVS Disks**.

Table 1-8 Common usage scenarios and suggestions

Common Usage Scenario	Suggestions
Resources are billed in pay-per-use mode.	● Top up your account in time to keep sufficient account balance. ● Pay attention to notifications about account arrears and top up your account in time.

1.9 Permissions

If you need to assign different permissions to employees in your enterprise to access your EVS resources, IAM is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you securely access your cloud resources.

With IAM, you can control access to specific cloud service resources. For example, if you want some resource management personnel in your enterprise to view EVS resources but do not want them to delete EVS resources or perform any other high-risk operations, you can grant permission to view EVS resources but not permission to delete them.

If your account does not require IAM for permissions management, you can skip this section.

IAM is a free service. You only pay for the resources in your account. For more information about IAM, see section "Service Overview" in the *Identity and Access Management User Guide*.

EVS Permissions

New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and attach policies or roles to these groups. The users then inherit permissions from the groups and can perform specified operations on cloud services based on the permissions they have been assigned.

EVS is a project-level service deployed for specific regions. To assign EVS permissions to a user group, specify the scope as region-specific projects and select a project for the permissions to take effect. If **All projects** is selected, the permissions will take effect for the user group in all region-specific projects. When accessing EVS, users need to switch to a region where they have been authorized to use EVS.

You can grant users permissions by using roles and policies.

- Roles: A type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. This mechanism provides only a limited number of service-level roles for authorization. When using roles to grant permissions, you need to also assign other roles on which the permissions depend to take effect. However, roles are not an ideal choice for fine-grained authorization and secure access control.
- Policies: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization, meeting requirements for secure access control. For example, you can grant ECS users only the permissions for managing a certain type of ECSs. Most policies define permissions based on APIs. For the API actions supported by EVS, see section "Permissions Policies and Supported Actions" in the *Elastic Volume Service API Reference*.

Table 1-9 lists all the system-defined roles and policies supported by EVS.

Table 1-9 System-defined roles and policies supported by EVS

Role/Policy Name	Description	Type	Dependency
EVS FullAccess	Full permissions for EVS. Users granted these permissions can create, attach, detach, query, and delete EVS resources, and expand capacity of EVS disks.	System-defined policy	None
EVS ReadOnlyAccess	Read-only permissions for EVS. Users granted these permissions can view EVS resource data only.	System-defined policy	None

Table 1-10 lists the common operations supported by each system-defined policy of EVS. Select the policies as required.

Table 1-10 Common operations supported by each system-defined policy of EVS

Operation	EVS FullAccess	EVS ReadOnlyAccess
Creating disks	√	x

Operation	EVS FullAccess	EVS ReadOnlyAccess
Viewing disk list	√	√
Viewing disk details	√	√
Attaching disks	√	x
Detaching disks	√	x
Deleting disks	√	x
Expanding disk capacities	√	x
Adding tags for disks	√	x
Modifying tags	√	x
Deleting tags	√	x
Searching for disks by tag	√	√
Changing disk names	√	x

Related Links

- Permissions management sections in the *Identify and Access Management User Guide*
- **Creating a User and Granting EVS Permissions**
- Section "Permissions Policies and Supported Actions" in the *Elastic Volume Service API Reference*

1.10 Constraints

This section describes the constraints on using EVS.

Table 1-11 Constraints on using EVS

Scenario	Item	Restrictions
Capacity of a data disk	• High I/O: 10 GiB to 32,768 GiB • General Purpose SSD: 10 GiB to 32,768 GiB • Ultra-high I/O: 10 GiB to 32,768 GiB • Extreme SSD: 10 GiB to 32,768 GiB	

Scenario	Item	Restrictions
Disk performance	Common I/O	• Maximum IOPS per disk: 1,000 • Maximum throughput per disk: 40 MiB/s
	Ultra-high I/O	• Maximum IOPS per disk: 20,000 • Maximum throughput per disk: 320 MiB/s
General Purpose SSD	• Maximum IOPS per disk: 20,000 • Maximum throughput per disk: 250 MiB/s	
Disk creation restrictions on accounts	Permission requirement	The account used to create EVS disks must have the evs:volumes:create permission. For how to add permissions, see **EVS Custom Policies**
Disk attachment	Constraints on region and AZ	The disk and server must be in the same region and AZ.
	Maximum number of servers that a non-shared disk can be attached to	1
	Maximum number of servers that a shared disk can be attached to	16
	Maximum number of disks that can be attached to an ECS	This value varies with ECS types. For details, see section "Can I Attach Multiple Disks to an ECS?" in the *Elastic Cloud Server User Guide*.
	Maximum number of disks that can be attached to a BMS	60 (1 system disk and 59 data disks)
	Device name	• System disk: /dev/vda, /dev/sda, and /dev/xvda • Data disk: /dev/vd[b-z], /dev/sd[b-z], and /dev/xvd[b-z]
Disk capacity expansion	Capacity expansion	Disk capacity can be expanded, but cannot be reduced.

Scenario	Item	Restrictions
	Capacity expansion of non-shared disks	Some server OSs support the capacity expansion of non-shared, In-use disks. For details, see **Expanding Capacity for an In-use EVS Disk**.
	Capacity expansion of shared disks	A shared disk must be detached from all its servers before expansion. That is, the shared disk status must be **Available**.
	Expansion increment	1 GiB
Disk detachment	System disk detachment	A system disk can only be detached offline, which means that the server must be in the **Stopped** state.
	Data disk detachment	A data disk can be detached online or offline, that is, its server can either be in the **Running** or **Stopped** state.
Disk deletion	Deletion of disks	• Pay-per-use disk: It can only be deleted when the following conditions are met: – The disk status is **Available**, **Error**, **Expansion failed**, **Restoration failed**, or **Rollback failed**. – The disk is not added to any replication pair in the Storage Disaster Recovery Service (SDRS). For any disk already added to a replication pair, you need to first delete the replication pair by referring to section "Deleting a Replication Pair" in the *Storage Disaster Recovery Service User Guide* and then delete the disk. – The disk is not locked by any service. – The shared disk has been detached from all its servers.
Tag	Maximum number of tags that can be added for a disk	10

1.11 EVS and Other Services

Figure 1-8 shows the relationships between EVS and other services.

Figure 1-8 Relationships between EVS and other services

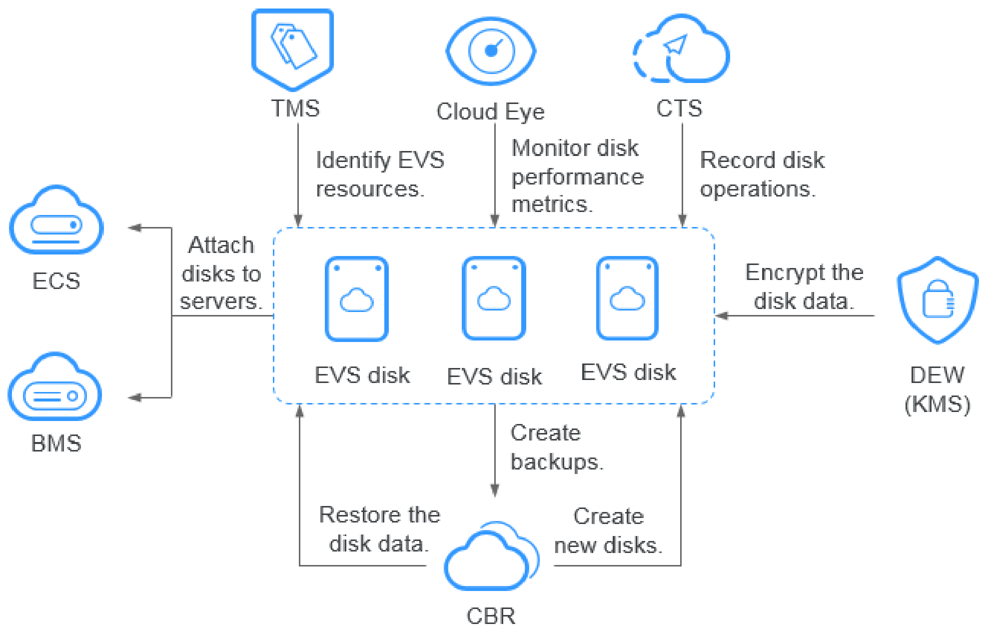


Table 1-12 EVS and other services

Interactive Function	Related Service	Reference
EVS disks can be attached to ECSs and used as scalable block storage devices.	ECS	• Attaching a Non-Shared Disk • Attaching a Shared Disk
SCSI EVS disks can be attached to BMSs and used as scalable block storage devices.	BMS	
Backups can be created for EVS disks to guarantee the reliability and security of the server data.	CBR	• EVS Backup • Managing EVS Backups
EVS encryption feature relies on KMS. You can use the keys provided by KMS to encrypt EVS disks (both system and data disks), thus improving EVS disk data security.	KMS	• EVS Encryption • Managing Encrypted EVS Disks
After EVS is enabled, the performance metrics of monitored disks can be viewed through Cloud Eye without installing any additional plug-in. The monitored metrics include Disk Read Rate, Disk Write Rate, Disk Read Requests, and Disk Write Requests.	Cloud Eye	Viewing EVS Monitoring Data

Interactive Function	Related Service	Reference
Cloud Trace Service (CTS) records operations of EVS resources, facilitating user query, audit, and backtracking.	CTS	**Auditing**
Tag Management Service (TMS) tags are used to identify EVS resources for purposes of easy categorization and quick search.	TMS	**Adding a Tag**

1.12 Basic Concepts

1.12.1 EVS Concepts

Table 1-13 EVS concepts

Concept	Description
IOPS	Number of read/write operations performed by an EVS disk per second
Throughput	Amount of data read from and written into an EVS disk per second
Read/write I/O latency	Minimum interval between two consecutive read/write operations of an EVS disk
VBD	A device type of EVS disks. VBD EVS disks only support basic SCSI read/write commands.
SCSI	A device type of EVS disks. SCSI EVS disks support transparent SCSI command transmission and allow the server OS to directly access the underlying storage media.

1.12.2 Region and AZ

Concept

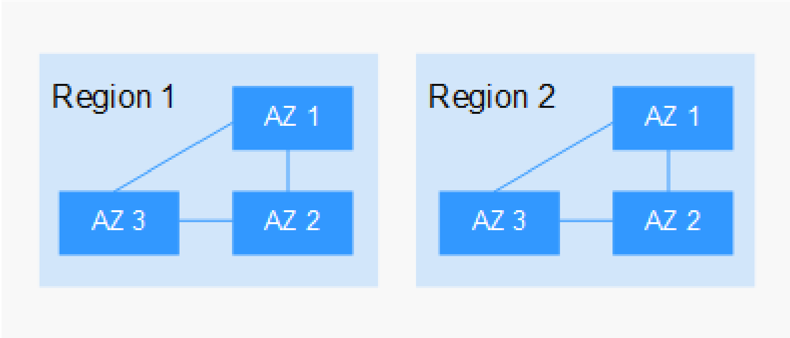
A region and availability zone (AZ) identify the location of a data center. You can create resources in a specific region and AZ.

- A region is a physical data center, which is completely isolated to improve fault tolerance and stability. The region that is selected during resource creation cannot be changed after the resource is created.
- An AZ is a physical location where resources use independent power supplies and networks. A region contains one or more AZs that are physically isolated

but interconnected through internal networks. Because AZs are isolated from each other, any fault that occurs in one AZ will not affect others.

Figure 1-9 shows the relationship between regions and AZs.

Figure 1-9 Regions and AZs



Selecting a Region

Select a region closest to your target users for lower network latency and quick access.

Selecting an AZ

When deploying resources, consider your applications' requirements on disaster recovery (DR) and network latency.

- For high DR capability, deploy resources in different AZs within the same region.
- For lower network latency, deploy resources in the same AZ.

Regions and Endpoints

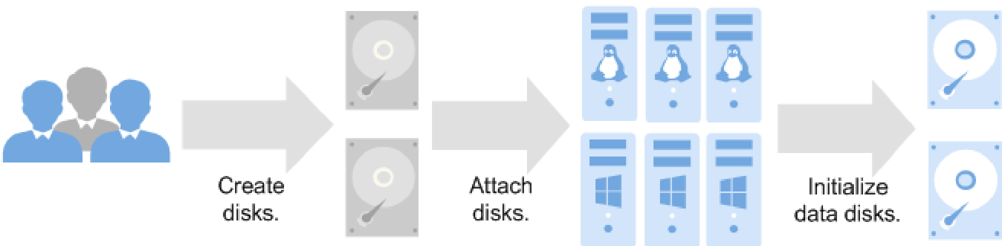
Before you use an API to call resources, specify its region and endpoint. For more details, see **Regions and Endpoints**.

2 Getting Started

2.1 Process Overview

Figure 2-1 shows the EVS process overview.

Figure 2-1 Process overview



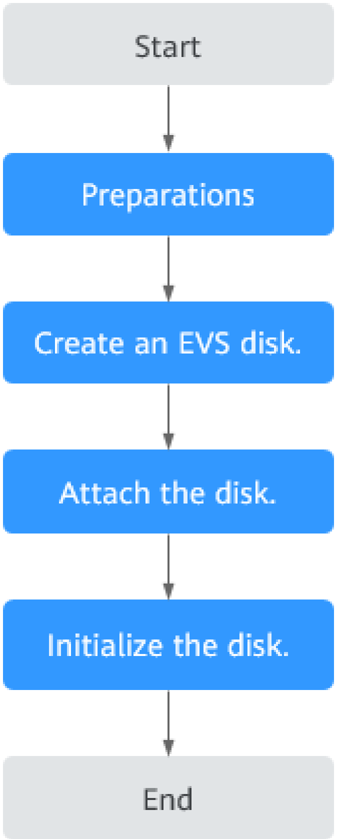
EVS disks can be attached to servers and be used as system disks or data disks. **Table 2-1** lists the disk purchasing methods.

Table 2-1 Method of creation

Disk	Description	Method
System disk	System disks are created along with servers and cannot be created separately.	• See section "Creating an ECS" in the *Elastic Cloud Server User Guide*. • See section "Creating a BMS" in the *Bare Metal Server User Guide*.
Data disk	Data disks can be created along with servers or separately.	• See section "Creating an ECS" in the *Elastic Cloud Server User Guide*. • See section "Creating a BMS" in the *Bare Metal Server User Guide*. • **Create an EVS Disk**

Figure 2-2 shows how to purchase a data disk separately.

Figure 2-2 Process overview



1. **Preparations**: Register an account on the console and obtain the permissions required to create servers and disks.
2. **Create a disk.** Configure the disk parameters, including the disk type, size, name, and other information. For details, see **Create an EVS Disk**.
3. **Attach the data disk.** Attach the separately created 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**
   - Windows
     - **Initializing a Windows Data Disk (Windows Server 2008)**
     - **Initializing a Windows Data Disk (Windows Server 2019)**
     - **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**
     - **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2012)**

- Linux
  - **Initializing a Linux Data Disk (fdisk)**
  - **Initializing a Linux Data Disk (parted)**
  - **Initializing a Linux Data Disk Larger Than 2 TiB (parted)**

2.2 Create an EVS Disk

Scenarios

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

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

Constraints

Table 2-2 Constraints on creating disks

Created On	Description
The EVS console	• Disks created on the EVS console are data disks. You need to manually attach them to servers. • Disks can only be attached to servers in the same region and AZ. Once created, the region and AZ cannot be changed. • 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 **Managing Quotas**.
The ECS console	• System disks can only be created together with servers and are automatically attached. • Data disks created together with servers or added after the server creation are automatically attached. • Disks will have the same billing mode as their server if the disks are created together with the server. • By default, disks created with ECSs are VBD disks, and those created with BMSs are SCSI disks.
-	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 **Create Disk**.

Step 4 Configure basic disk information according to **Table 2-3**.

Table 2-3 Disk parameters

Parameter	Description	Example Value
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. **NOTE** • Disks can only be attached to the servers in the same AZ. • The AZ of a disk cannot be changed after the disk has been created.	-
Disk Specifications	Mandatory • Common I/O • Ultra-high I/O • General Purpose SSD • Extreme SSD	Common I/O
Disk Size (GiB)	Mandatory 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. **NOTE** • 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.	20 GiB

<table>
<tr><th>Parameter</th><th>Description</th><th>Example Value</th></tr>
<tr><td>Create from backup</td><td>Optional
Specifies to create the disk from a backup.
Click Select Data Source and choose Create from backup. On the displayed page, select the target backup and click OK.
NOTE
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.
If a disk is created from a backup of a system disk, the new disk can be used as a data disk only.</td><td>autobackup-001</td></tr>
<tr><td>Automatic Backup</td><td>CBR lets you back up EVS disks and ECSs and use the backups to restore data. After you configure automatic backup, the system will associate the EVS disk with the backup vault and apply the selected policy to the vault to periodically back up the disk.
Do not use: Skip this configuration if backup is not required. If you need backup protection after a disk has been created, log in to the CBR console, locate the desired vault, and associate the disk with the vault.
Use existing:
Vault: Select an existing vault from the drop-down list.
Backup Policy: Select a backup policy from the drop-down list, or go to the CBR console and configure a desired one.
Buy new:
Vault Name: Enter a vault name, which can contain a maximum of 64 characters, including letters, digits, periods (.), underscores (_), and hyphens (-), for example, vault-f61e. The default naming rule is vault_xxxx.
Vault Capacity: Enter a vault capacity for storing disk backups. The vault capacity cannot be less than the size of the disk to be backed up. The value ranges from the disk size to 10,485,760 in the unit of GiB.
Backup Policy: Select a backup policy from the drop-down list, or go to the CBR console and configure a desired one.</td><td>Do not use</td></tr>
</table>

Parameter	Description	Example Value
Share	Optional ● If **Share** is not selected, a common disk is created. ● If **Share** is selected, a shared disk is created, and the shared disk can be attached to multiple servers. If you select both **SCSI** and **Share**, a shared SCSI disk is created. **NOTE** The sharing attribute of a disk cannot be changed after the disk has been created. For details about shared EVS disks, see **Managing Shared EVS Disks**.	-
SCSI	Optional ● If you do not select **SCSI**, a VBD disk is created. VBD is the default device type of EVS disks. ● If you select **SCSI**, 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. **NOTE** The device type of a disk cannot be changed after the disk has been created. For details about the ECS types, OSs, and ECS software supported by SCSI EVS disks, see **Device Types and Usage Instructions**.	-

Paramete r	Description	Example Value
Encryption	Optional Disk encryption is used for data disk encryption only. System disk encryption relies on the image. For details, see the *Image Management Service User Guide*. To use the disk encryption function, select **Encryption**. The displayed dialog box contains the following parameters: ● 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 KMS access rights are not granted to EVS, the **Create Agency** dialog box will be displayed. Otherwise, it will not be displayed. Click **Yes** 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. After the KMS access rights have been granted, follow-up operations do not require the rights to be granted again. ● KMS Key Name **NOTE** **KMS Key Name** is displayed only after the KMS access rights have been granted. For details, see "Create Agency" above. **KMS Key Name** is the identifier of the key, and you can use **KMS Key Name** to specify a KMS key and use it for encryption. One of the following keys can be used: – Default Master Key: After the KMS access rights have been granted to EVS, the system automatically creates a Default Master Key **evs/default**. – CMKs: Existing or newly created CMKs. For details, see **Management** > **Creating a CMK** in the *Key Management Service User Guide*.	-

Parameter	Description	Example Value
	NOTE ● 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. ● The encryption attribute of a disk cannot be changed after the disk has been created. For details, see **EVS Encryption**.	
Tag	Optional During the EVS disk creation, you can tag the EVS resources. Tags identify cloud resources for purposes of easy categorization and quick search. A tag is composed of a key-value pair. ● Key: Mandatory if the disk is going to be tagged A tag key can contain a maximum of 36 characters, including letters, digits, and underscores (_). ● Value: Optional if the disk is going to be tagged A tag value can contain a maximum of 43 characters, including letters, digits, underscores (_), periods (.), and hyphens (-). **NOTE** ● A maximum of 10 tags can be added for an EVS disk. ● Tag keys of the same EVS disk must be unique. ● Except for tagging the disk during disk creation, you can also add, modify, or delete tags for existing disks. For details, see **Managing a Tag**. For details about tags, see the *Tag Management Service User Guide*.	-
Disk Name	Mandatory ● If you create disks individually, this parameter value is used as the actual disk name. The name can contain a maximum of 64 characters. ● 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.	For example, if you create two disks and set **volume** for **Disk Name**, the EVS disk names will be **volume-0001** and **volume-0002**.

Parameter	Description	Example Value
Quantity	Optional The number of disks to be created. The default value is set to **1**, which means only one disk is created. Currently, you can create up to 100 disks at a time. **NOTE** ● If the disk is created from a backup, batch creation is not possible, and this parameter must be set to **1**.	1

Step 5 Click **Create Now**.

Step 6 On the **Details** page, check the disk details.

- If you do not need to modify the specifications, click **Submit**.
- If you need to modify the specifications, click **Previous**.

Step 7 In the disk list, view the disk status.

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

----End

2.3 Attach an EVS Disk

2.3.1 Attaching a Non-Shared Disk

Scenarios

Separately created 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**.

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 shared disk can be attached only when the servers' statuses are **Running** or **Stopped**.
- A frozen disk cannot be attached.

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*.

Step 5 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 6 Initialize the disk.

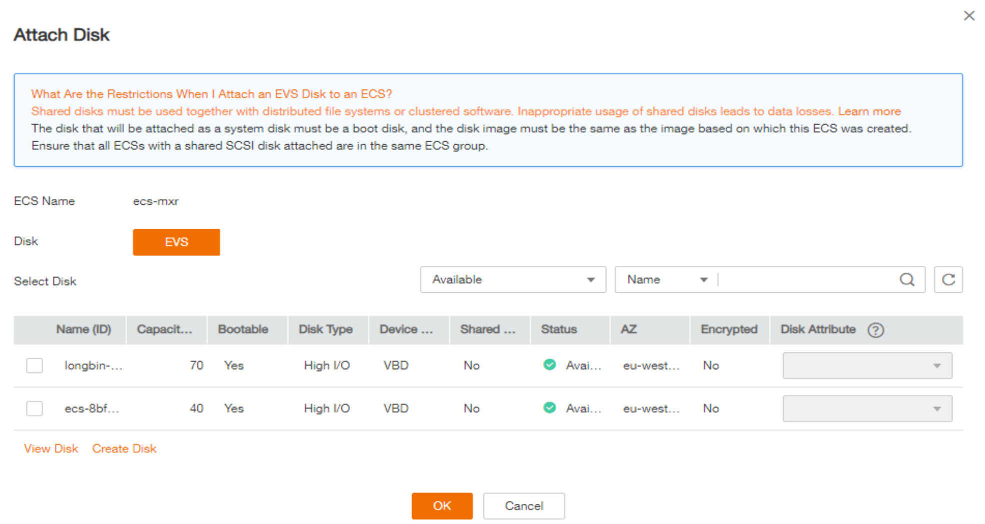
After the disk has been attached to a server, the disk can be used only after you have initialized it. For details, see **Introduction to Data Disk Initialization Scenarios and Partition Styles**.

----End

Attaching the Disk on the ECS Console

1. Log in to the management console.
2. Under **Computing**, choose **Elastic Cloud Server**.
3. In the search box above the upper right corner of the ECS list, enter the ECS name, IP address, or ID for search.
4. Click the name of the target ECS.
   The page providing details about the ECS is displayed.
5. Click the **Disks** tab. Then, click **Attach Disk**.
   The **Attach Disk** dialog box is displayed.

Figure 2-3 Attach Disk



6. 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 FAQ "What Are the Requirements for Attaching an EVS Disk to an ECS?" in the *Elastic Cloud Server User Guide*.

7. 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 I Attach My Disk to a Server?**.

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?**.

2.3.2 Attaching a Shared Disk

Scenarios

Separately created 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.

Prerequisites

- The shared disk status is **In-use** or **Available**.

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.

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 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.

Step 5 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. Under **Compute**, click **Elastic Cloud Server**.
3. In the search box above the upper right corner of the ECS list, enter the ECS name, IP address, or ID for search.
4. Click the name of the target ECS.

   The page providing details about the ECS is displayed.
5. Click the **Disks** tab. Then, click **Attach Disk**.

   The **Attach Disk** page is displayed.
6. 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 the system disk or a data disk but cannot specify a specific 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 FAQ "What Are the Requirements for Attaching an EVS Disk to an ECS?" in the *Elastic Cloud Server User Guide*.
7. 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 I Attach My Disk to a Server?**.

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?**.

2.4 Initialize an EVS Data Disk

2.4.1 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.

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.

Disk Partition Styles

Table 2-4 lists the common disk partition styles. In Linux, different partition styles require different partitioning tools.

Table 2-4 Disk partition styles

Disk Partition Style	Maximum Disk Capacity Supported	Maximum Number of Partitions Supported	Linux Partitioning Tool
Master Boot Record (MBR)	2 TiB	• 4 primary partitions • 3 primary partitions and 1 extended partition 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: • 3 primary partitions and 1 extended partition, with the extended partition divided into 3 logical partitions • 1 primary partition and 1 extended partition, with the extended partition divided into 5 logical partitions	• fdisk • parted
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 Windows Data Disk (Windows Server 2008)**
- **Initializing a Windows Data Disk (Windows Server 2019)**

- **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 Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**
- **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2012)**
- **Initializing a Linux Data Disk Larger Than 2 TiB (parted)**

2.4.2 Initializing a Windows Data Disk (Windows Server 2008)

Scenarios

This section uses Windows Server 2008 R2 Enterprise 64bit to describe how to initialize a data disk attached to a server running Windows.

The maximum disk capacity supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Therefore, use the GPT partition style if your disk capacity is larger than 2 TiB. For details, see **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**. To learn more about disk partition styles, 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 the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Procedure

Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

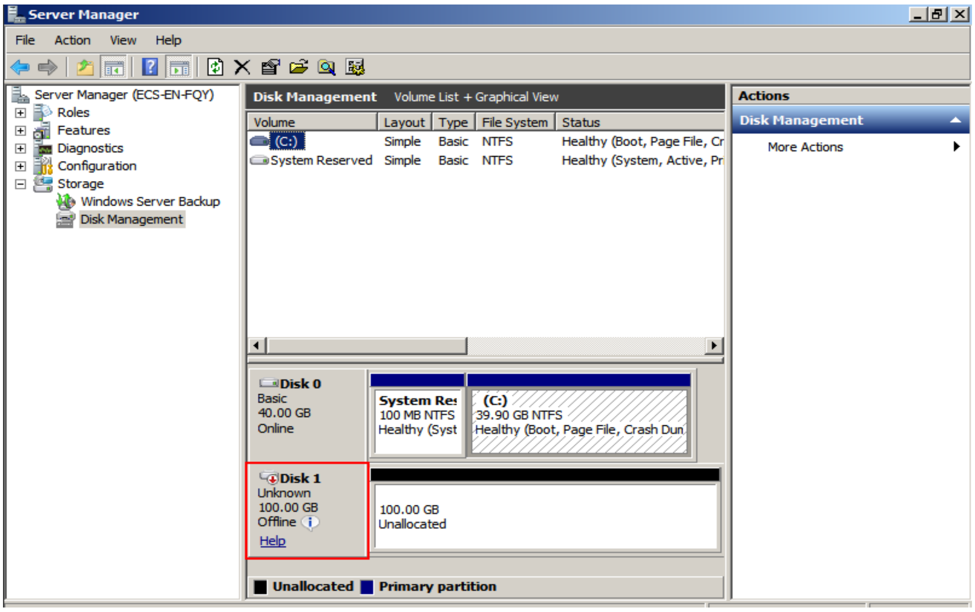
The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The **Disk Management** window is displayed.

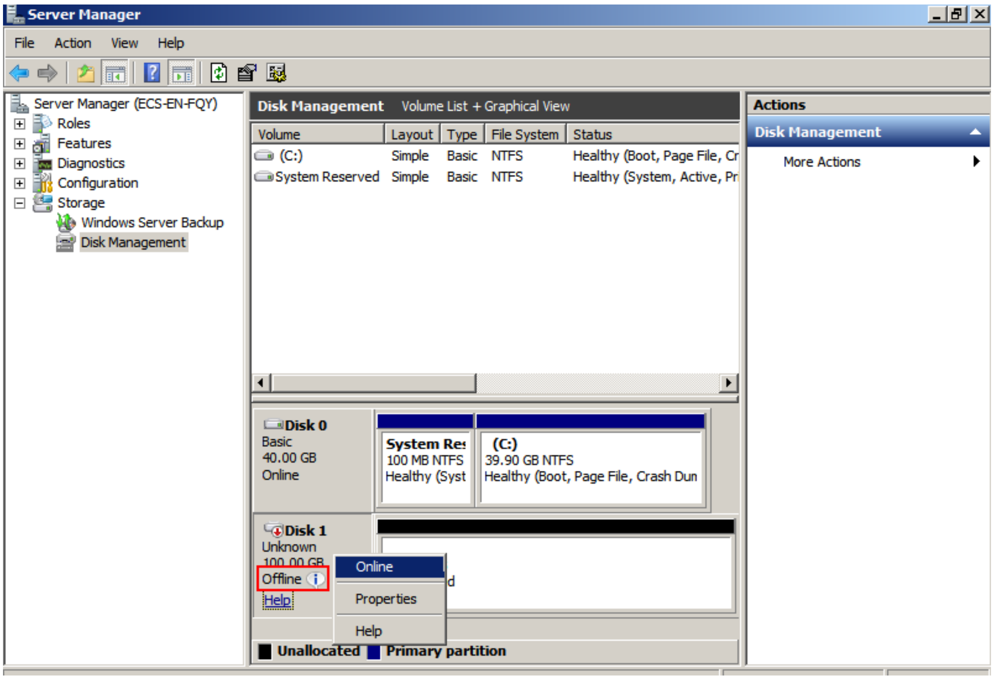
- If **Figure 2-4** is displayed, the new disk is offline. Go to **Step 3**.
- If **Figure 2-7** is displayed, the **Initialize Disk** window is prompted. Go to **Step 5**.

Figure 2-4 Disk Management



Step 3 Disks are displayed in the right pane. In the **Disk 1** area, right-click **Offline** and choose **Online** from the shortcut menu to online the disk.

Figure 2-5 Online the disk

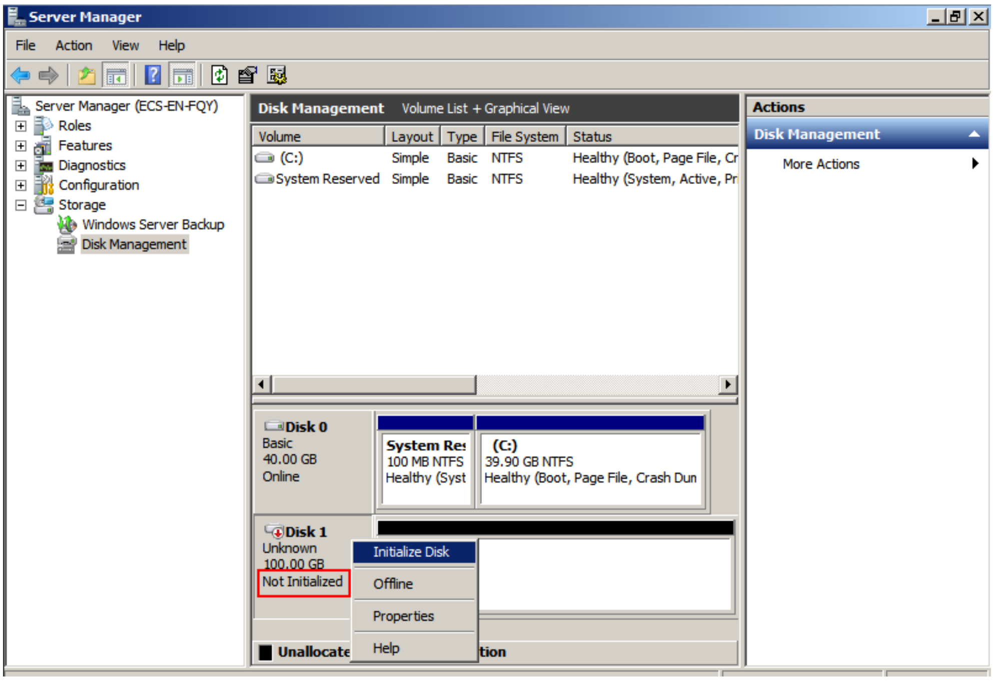


NOTE

If the disk is offline, you need to bring it online before initializing it.

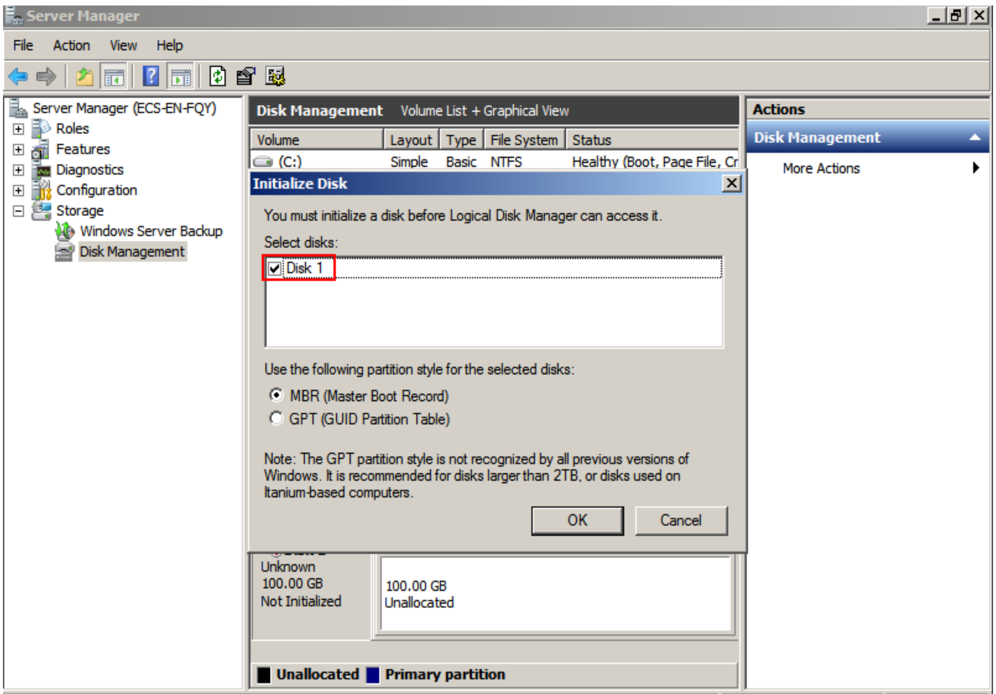
Step 4 After making the disk online, the disk status changes from **Offline** to **Not Initialized**. Right-click the disk status and choose **Initialize Disk** from the shortcut menu.

Figure 2-6 Initialize Disk



Step 5 In the **Initialize Disk** dialog box, select the target disk, click **MBR (Master Boot Record)** or **GPT (GUID Partition Table)**, and click **OK**.

Figure 2-7 Unallocated space



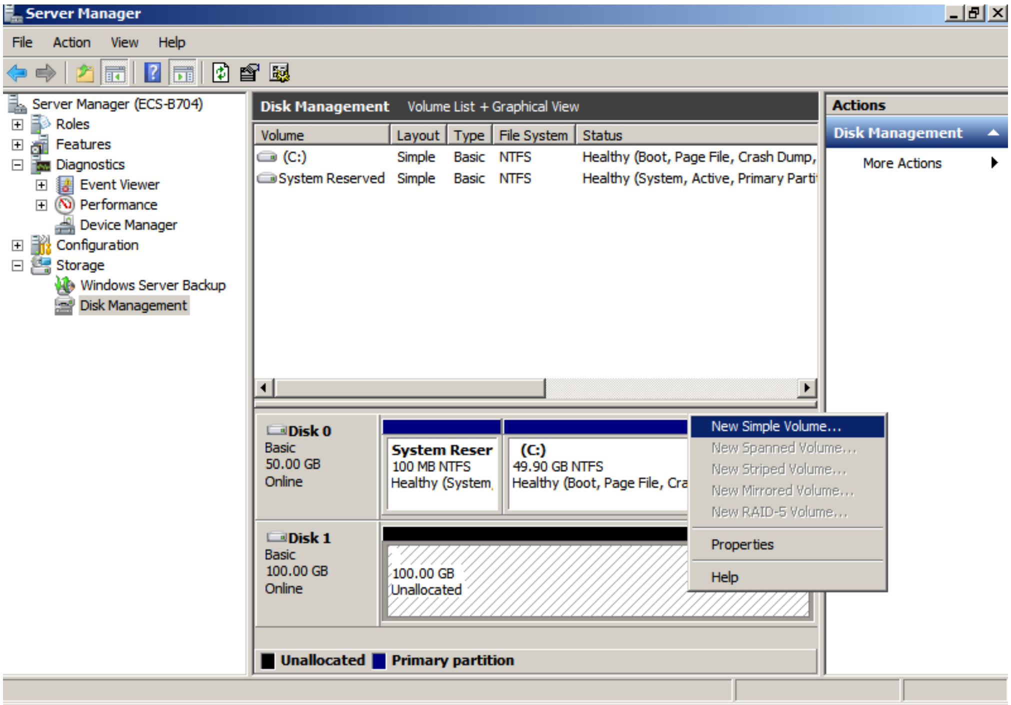
NOTICE

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.

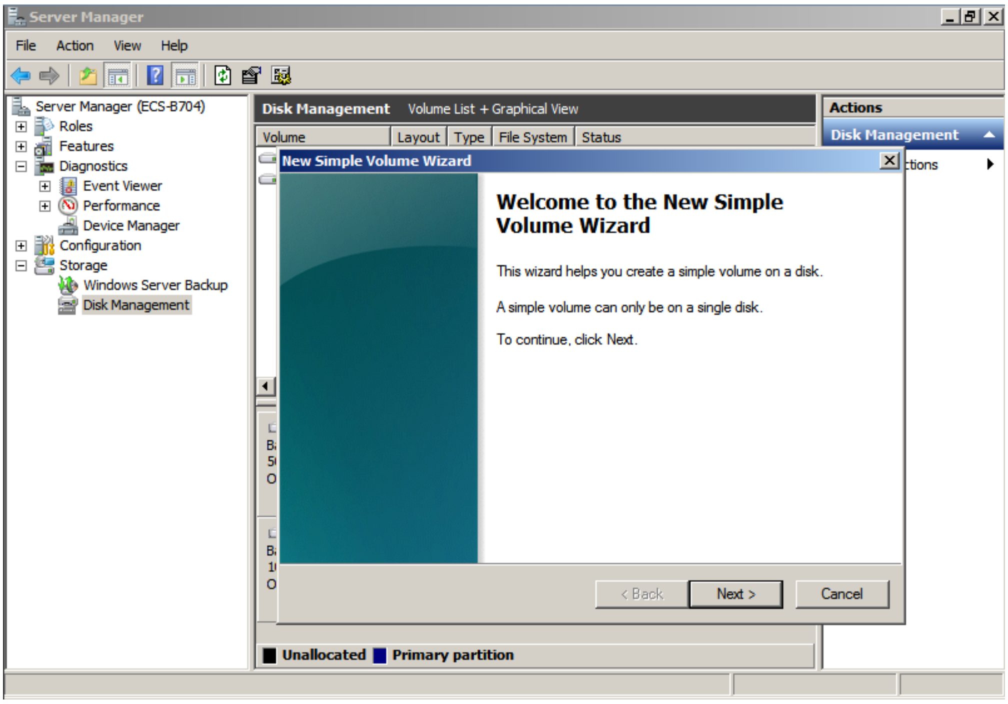
Step 6 Right-click at the unallocated space and choose **New Simple Volume** from the shortcut menu.

Figure 2-8 New Simple Volume



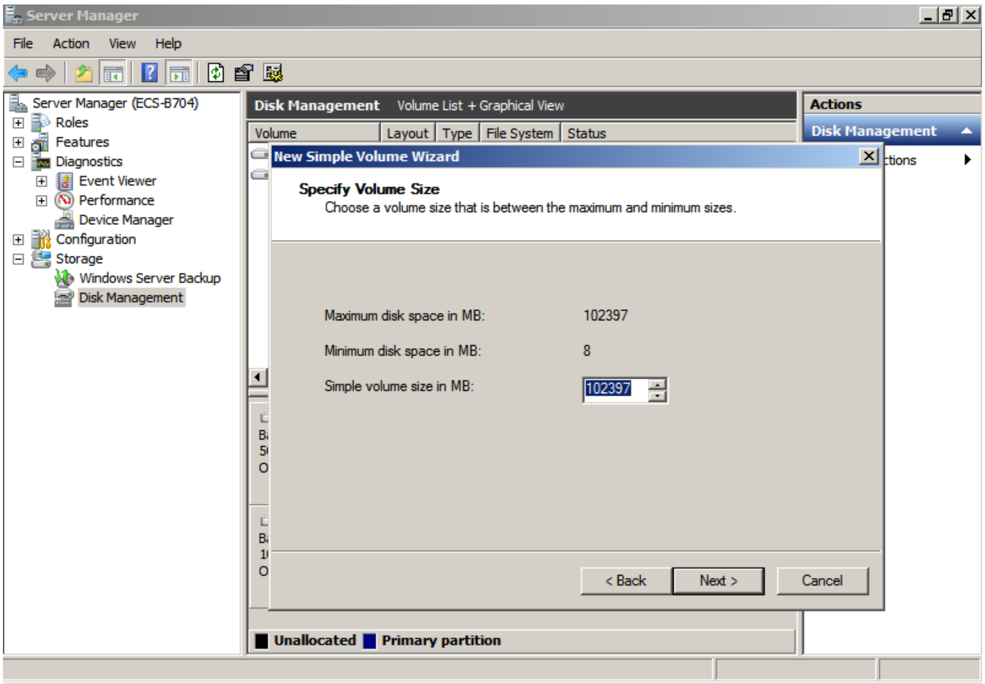
Step 7 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 2-9 New Simple Volume Wizard



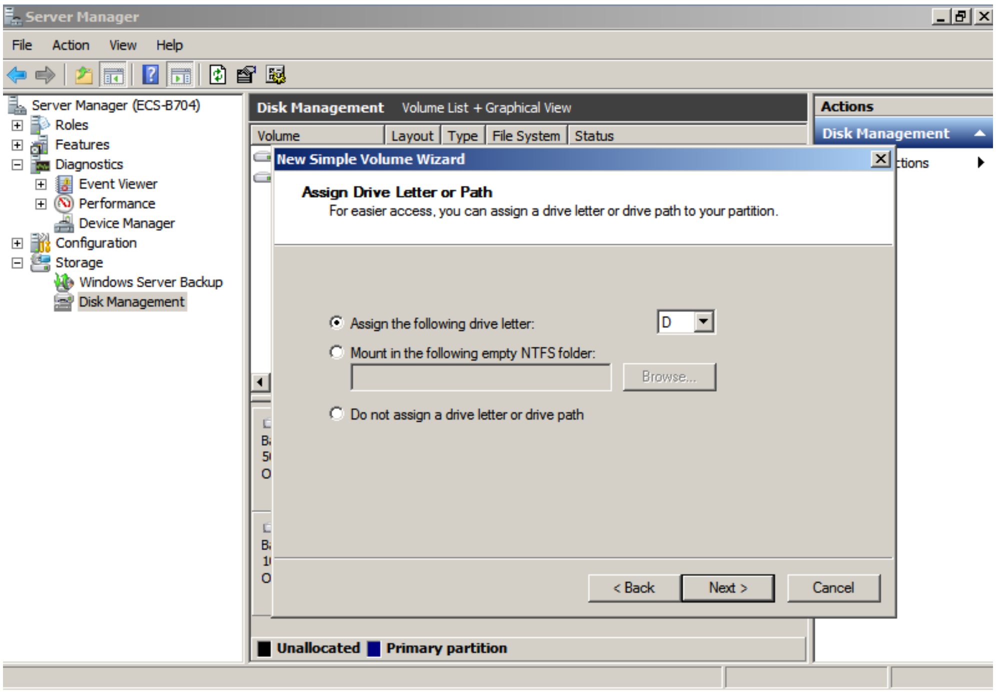
Step 8 Specify the volume size and click **Next**. The default value is the maximum size.

Figure 2-10 Specify Volume Size



Step 9 Assign the drive letter and click **Next**.

Figure 2-11 Assign Drive Letter or Path



Step 10 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

Figure 2-12 Format Partition

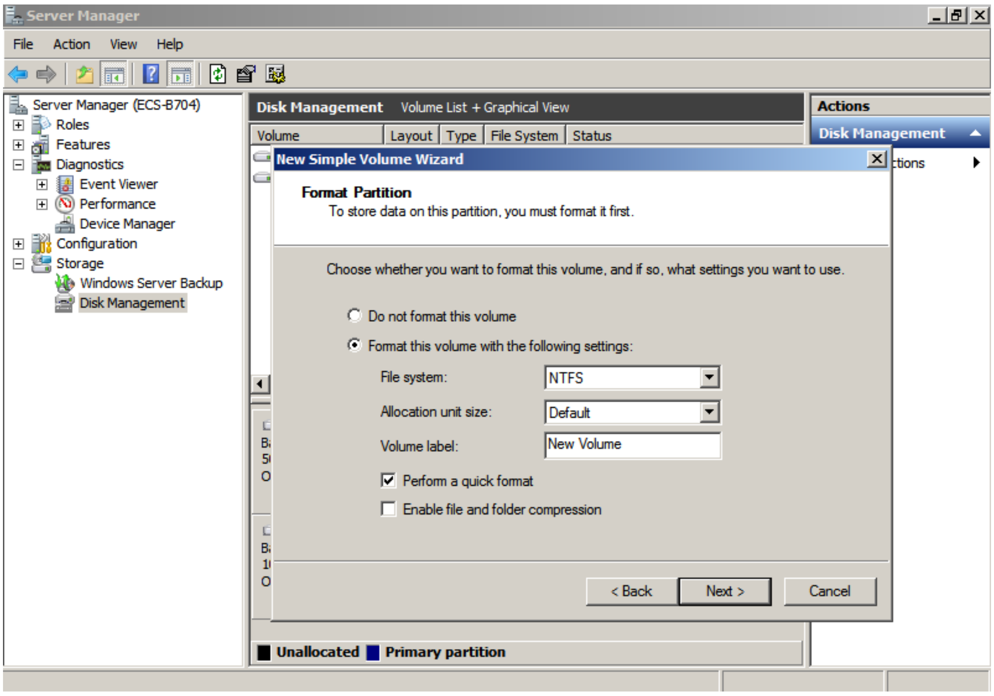
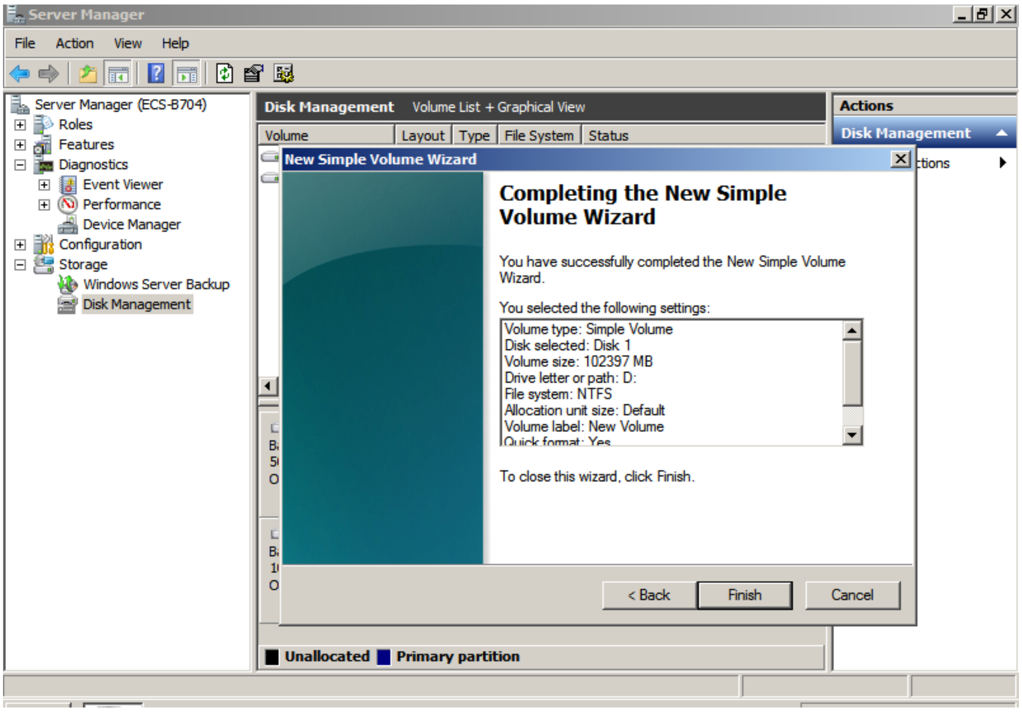


Figure 2-13 Completing the partition creation

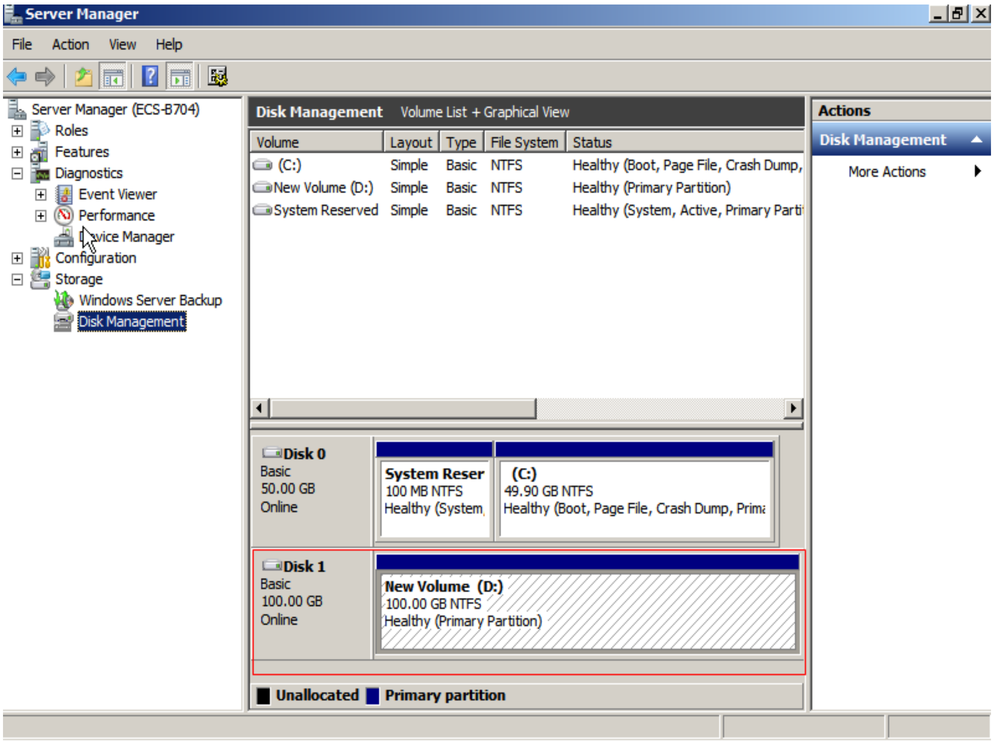


NOTICE

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

Step 11 Click **Finish**. Wait for the initialization to complete. When the volume status changes to **Healthy**, the initialization has finished successfully.

Figure 2-14 Disk initialization succeeded



----End

2.4.3 Initializing a Windows Data Disk (Windows Server 2019)

Scenarios

This section uses Windows Server 2019 Standard 64bit to describe how to initialize a data disk attached to a server running Windows.

The maximum disk capacity supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Therefore, use the GPT partition style if your disk capacity is larger than 2 TiB. For details, see **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**. To learn more about disk partition styles, 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

- You have attached a data disk to a server but not initialized the disk.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Procedure

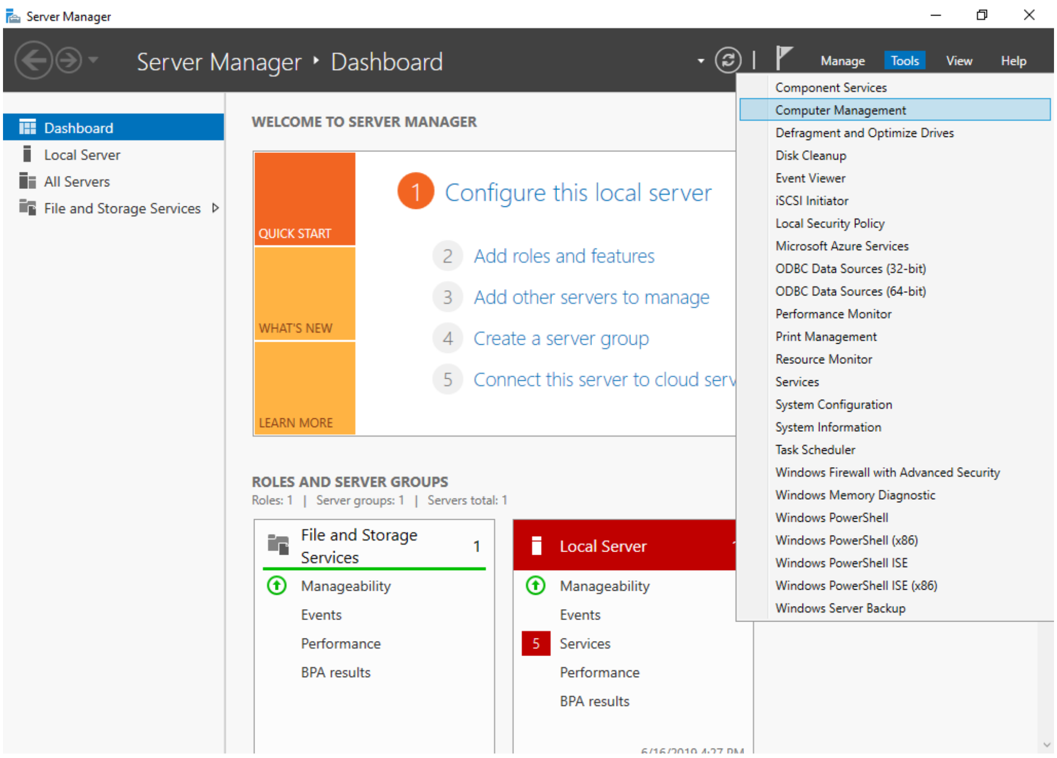
Step 1 On the desktop of the server, click the start icon in the lower left corner.

The **Windows Server** window is displayed.

Step 2 Click **Server Manager**.

The **Server Manager** window is displayed.

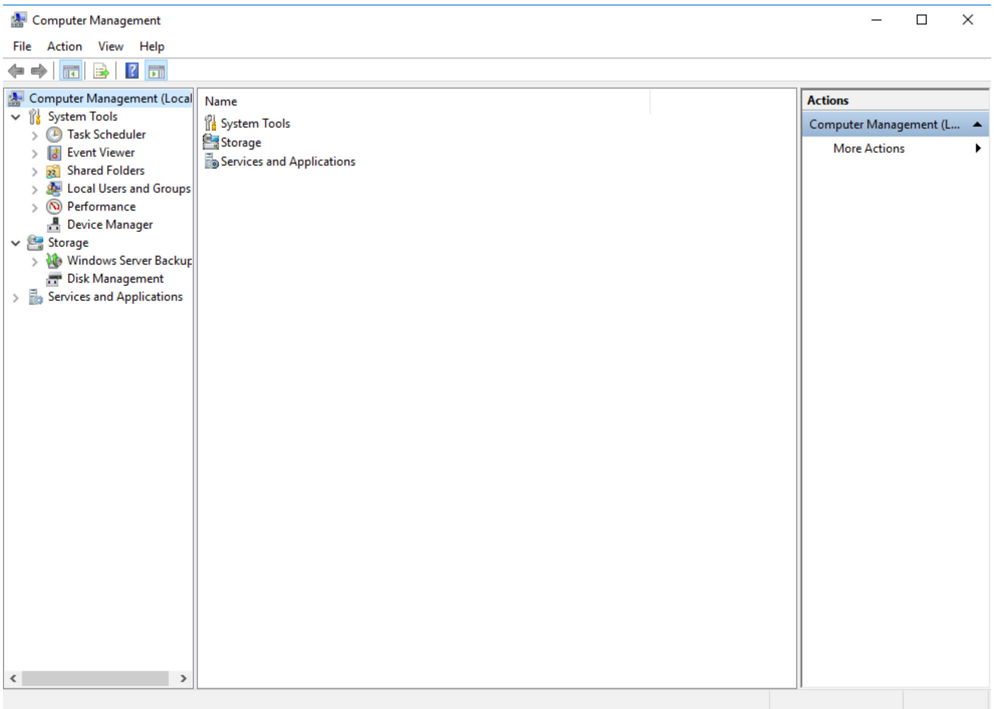
Figure 2-15 Server Manager



Step 3 In the upper right corner, choose **Tools** > **Computer Management**.

The **Computer Management** window is displayed.

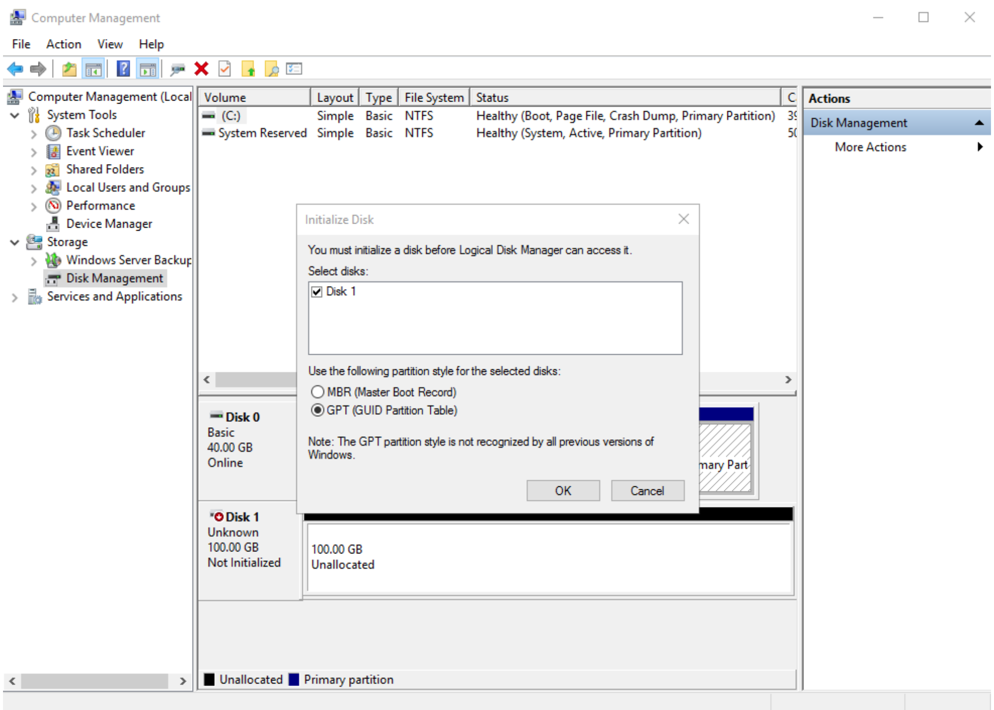
Figure 2-16 Computer Management



Step 4 Choose **Storage** > **Disk Management**.

Disks are displayed in the right pane. If there is a disk that is not initialized, the system will prompt you with the **Initialize Disk** dialog box.

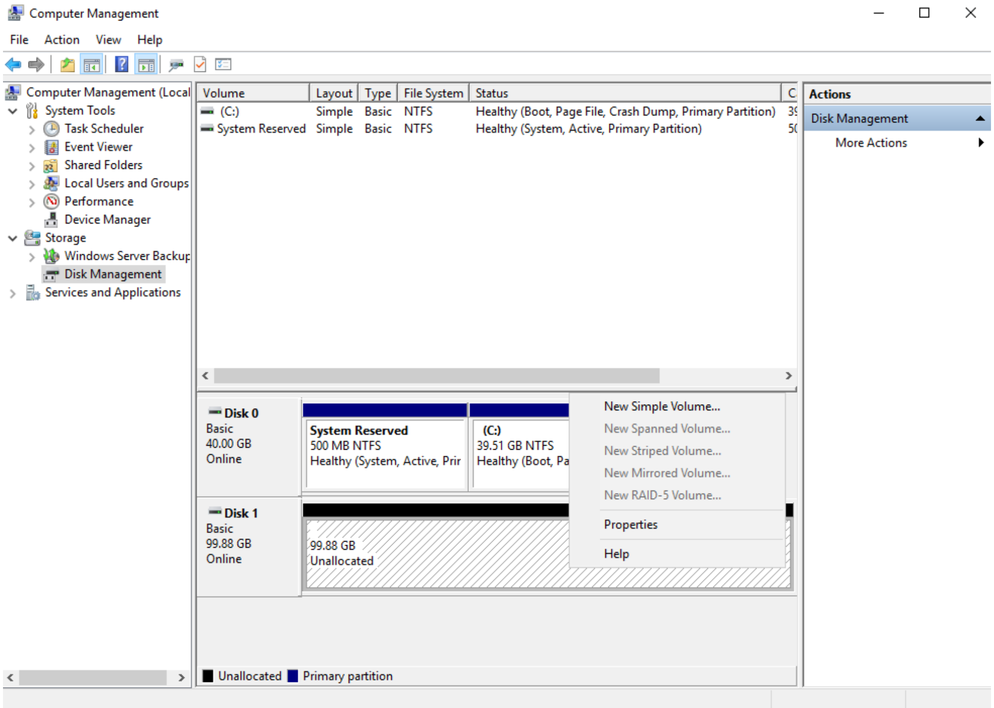
Figure 2-17 Disk list



Step 5 In the **Initialize Disk** dialog box, the to-be-initialized disk is selected. Select a disk partition style and click **OK**. In this example, **GPT (GUID Partition Table)** is selected.

The **Computer Management** window is displayed.

Figure 2-18 Computer Management



NOTICE

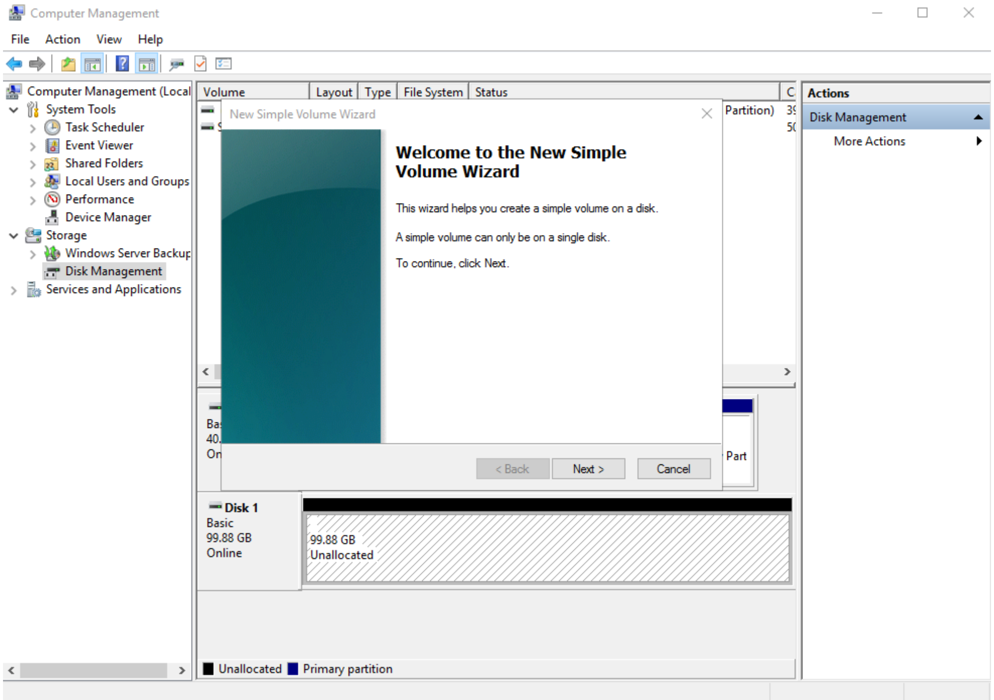
The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

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 6 Right-click at the unallocated disk space and choose **New Simple Volume** from the shortcut menu.

The **New Simple Volume Wizard** window is displayed.

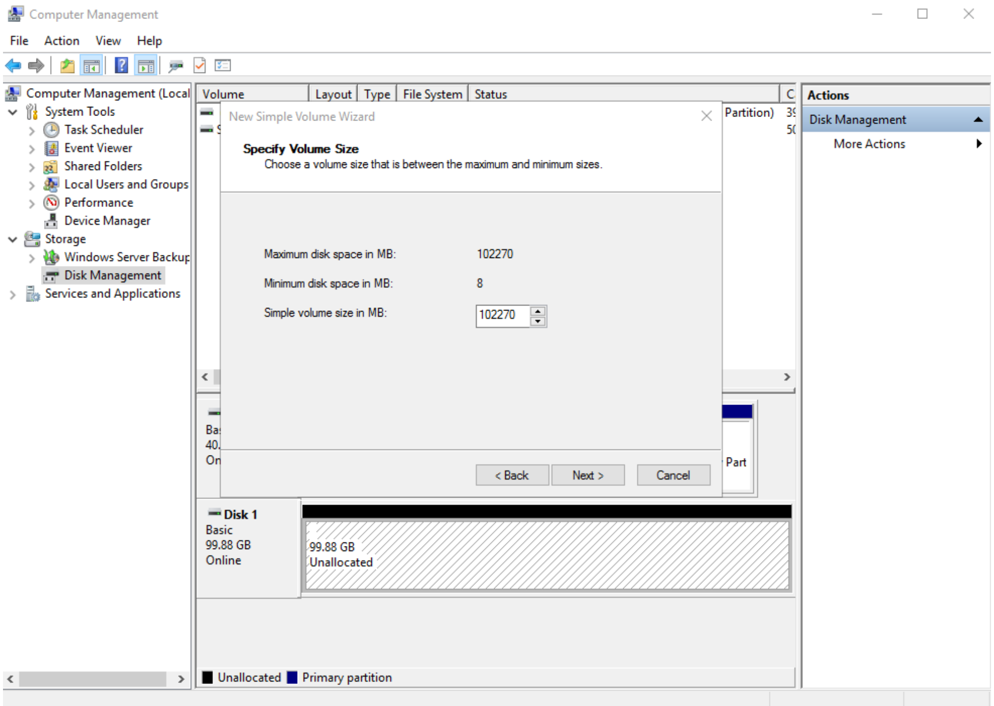
Figure 2-19 New Simple Volume Wizard



Step 7 Follow the prompts and click **Next**.

The **Specify Volume Size** page is displayed.

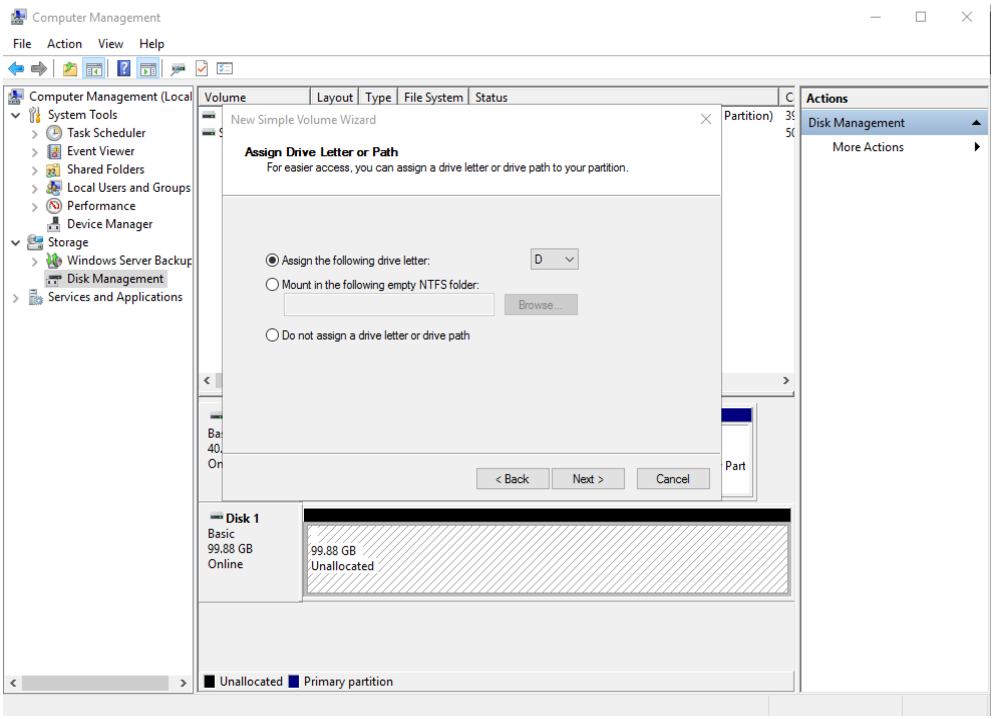
Figure 2-20 Specify Volume Size



Step 8 Specify the volume size and click **Next**. The system selects the maximum volume size by default. You can specify the volume size as required. In this example, the default setting is used.

The **Assign Drive Letter or Path** page is displayed.

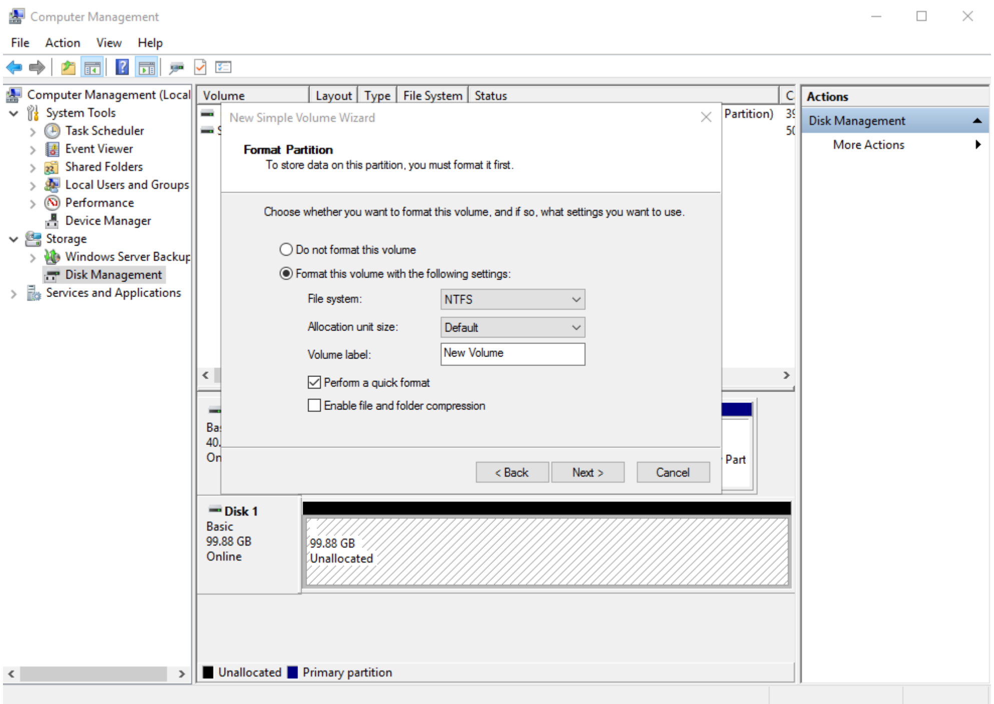
Figure 2-21 Assign Drive Letter or Path



Step 9 Assign a drive letter or path to your partition and click **Next**. The system assigns drive letter D by default. In this example, the default setting is used.

The **Format Partition** page is displayed.

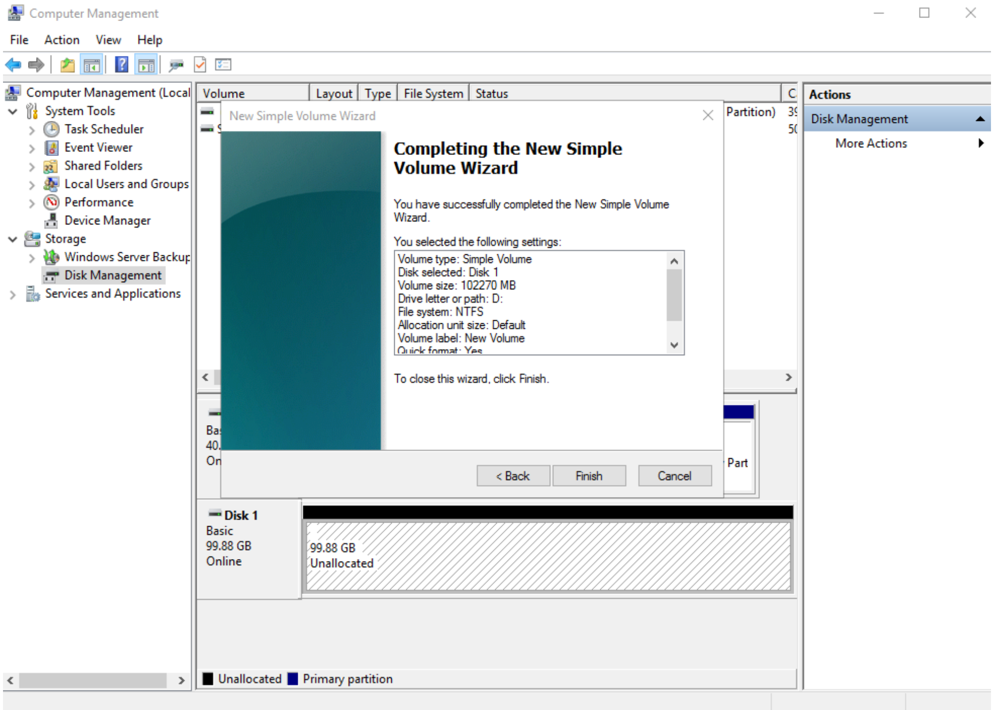
Figure 2-22 Format Partition



Step 10 Specify format settings and click **Next**. The system selects the NTFS file system by default. You can specify the file system type as required. In this example, the default setting is used.

The **Completing the New Simple Volume Wizard** page is displayed.

Figure 2-23 Completing the New Simple Volume Wizard



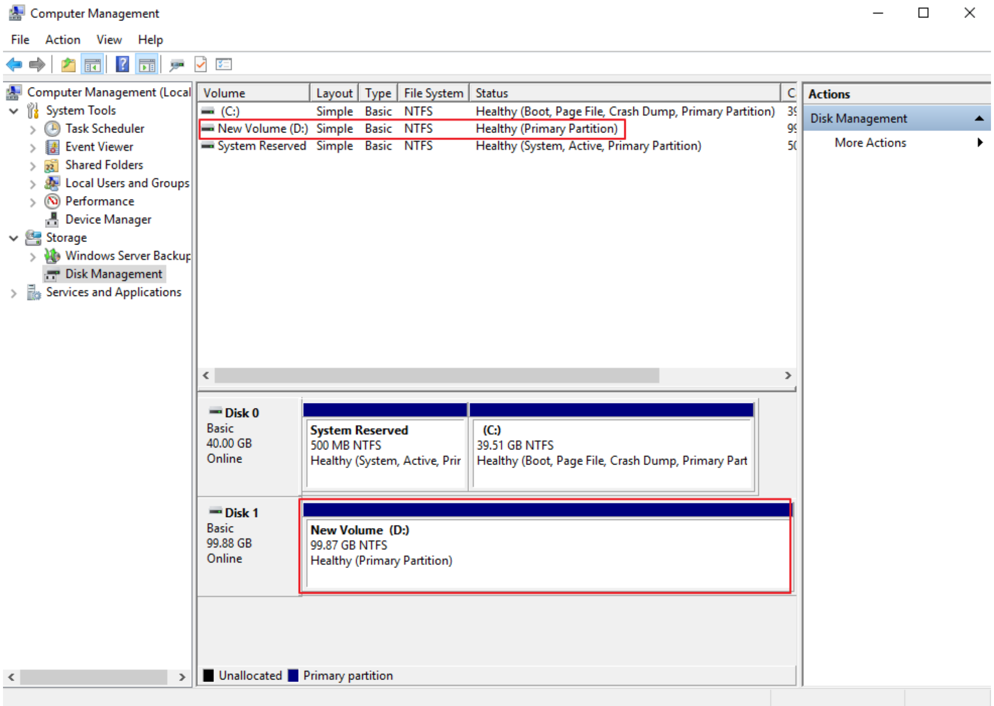
NOTICE

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

Step 11 Click **Finish**.

Wait for the initialization to complete. When the volume status changes to **Healthy**, the initialization has finished successfully.

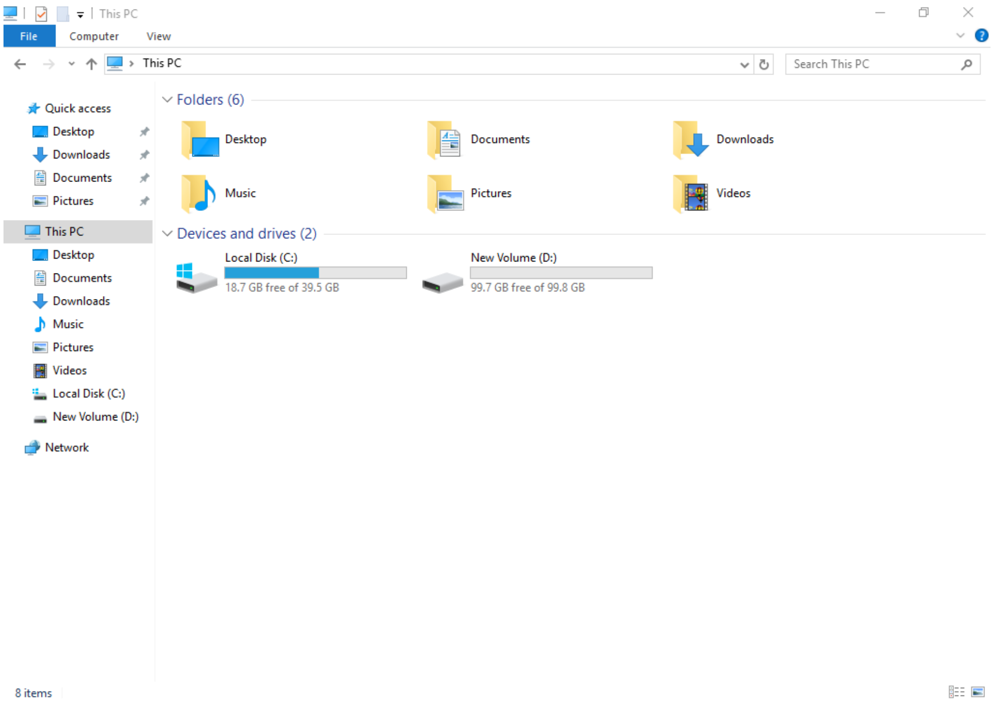
Figure 2-24 Disk initialized



Step 12 After the volume is created, click  on the task bar and check whether a new volume appears in **This PC**. In this example, New Volume (D:) is the new volume.

If New Volume (D:) appears, the disk is successfully initialized and no further action is required.

Figure 2-25 This PC



----End

2.4.4 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

- You have attached a data disk to a server but not initialized the disk.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

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 MBR is used, a maximum of four primary partitions, or three primary partitions plus one extended partition can be created. The extended partition 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. You can modify the **/etc/fstab** file to configure automount at startup. 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 server stop or start. This can even prevent the server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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

2.4.5 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

- You have attached a data disk to a server but not initialized the disk.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

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. You can modify the **/etc/fstab** file to configure automount at startup. 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 server startup. You can configure the **fstab** file of a server that has data. This operation will not affect the existing data.

The following example uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names (like **/dev/vdb1**) 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 server stop or start. This can even prevent your server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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

2.4.6 Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)

Scenarios

This section uses Windows Server 2008 R2 Standard 64bit to describe how 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 disk capacity supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Therefore, use the GPT partition style if your disk capacity is larger than 2 TiB. For details, see **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**. To learn more about disk partition styles, 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

- You have attached a data disk to a server but not initialized the disk.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Procedure

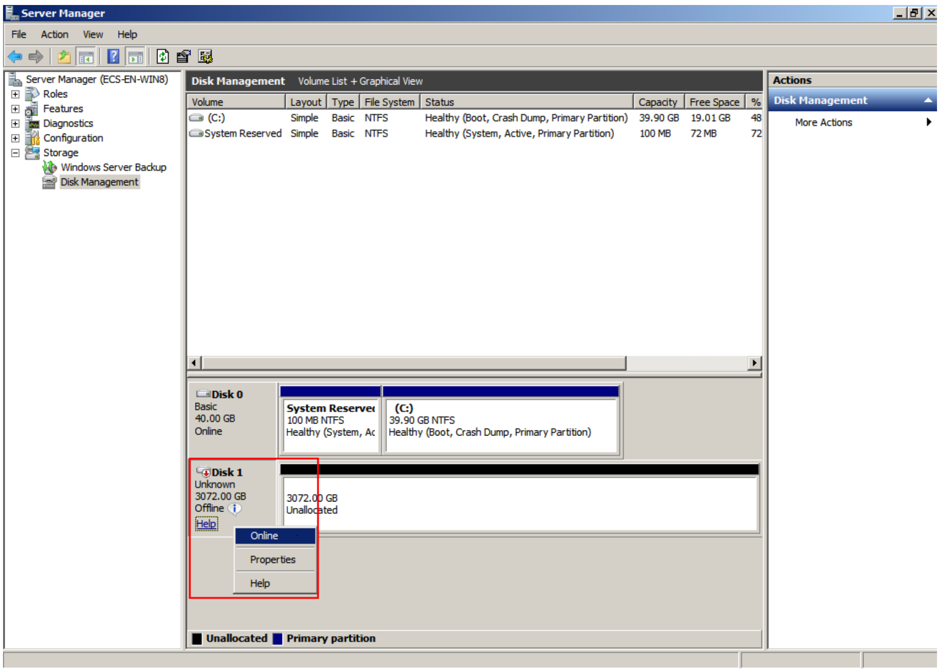
Step 1 On the desktop of the server, click **Start**.

The **Start** window is displayed.

Step 2 Right-click **Computer** and choose **Manage** from the short-cut menu.

The **Server Manager** window is displayed.

Figure 2-26 Server Manager (Windows Server 2008)

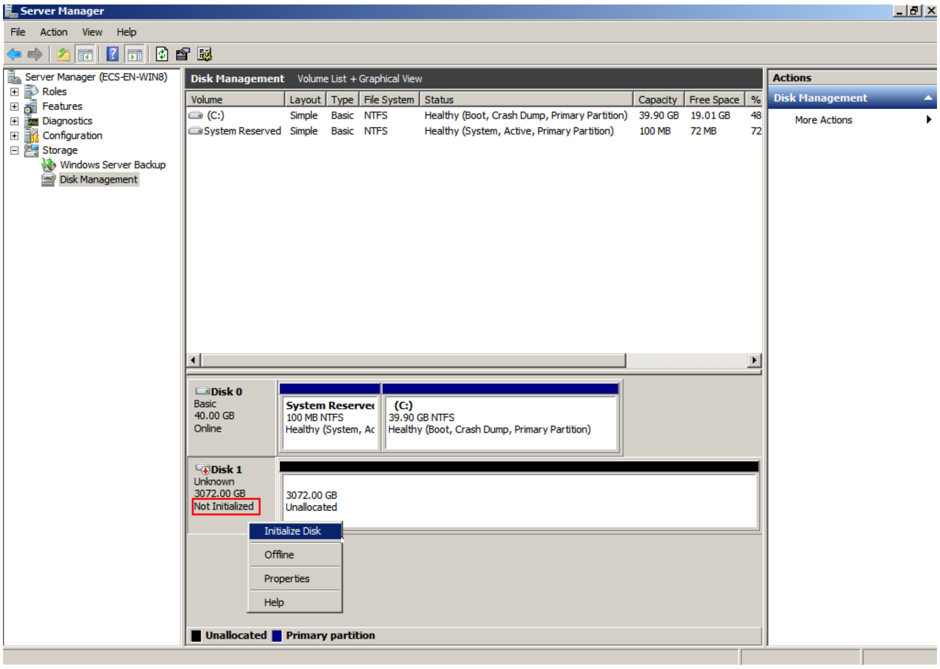


Step 3 Disks are listed in the right pane. If the new disk is offline, bring it online before initializing it.

In the **Disk 1** area, right-click and choose **Online** from the shortcut menu.

When the status of Disk 1 changes from **Offline** to **Not Initialized**, the disk has been brought online.

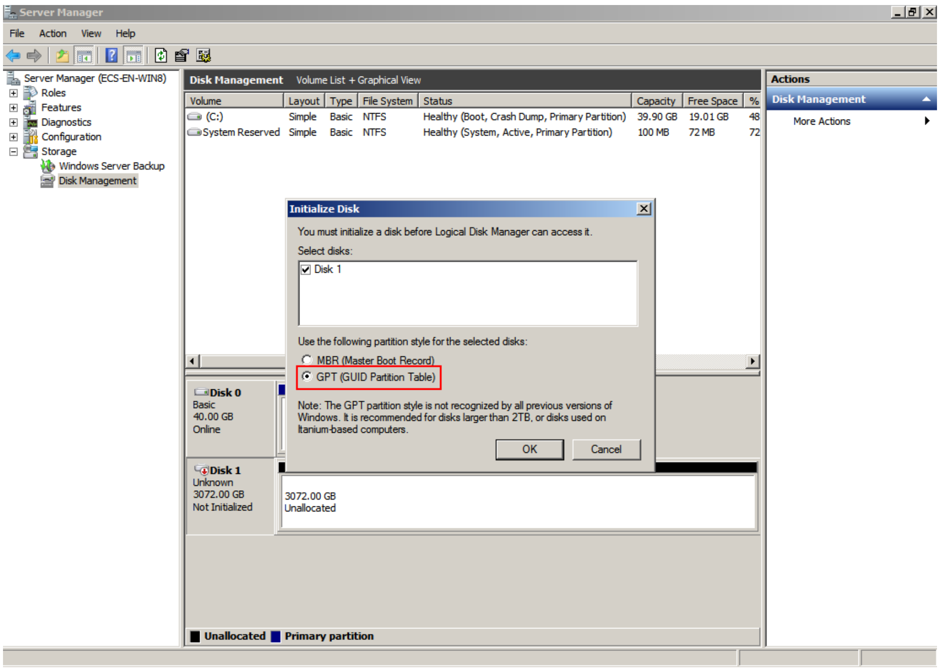
Figure 2-27 Bring online succeeded (Windows Server 2008)



Step 4 In the **Disk 1** area, right-click and choose **Initialize Disk** from the shortcut menu.

The **Initialize Disk** dialog box is displayed.

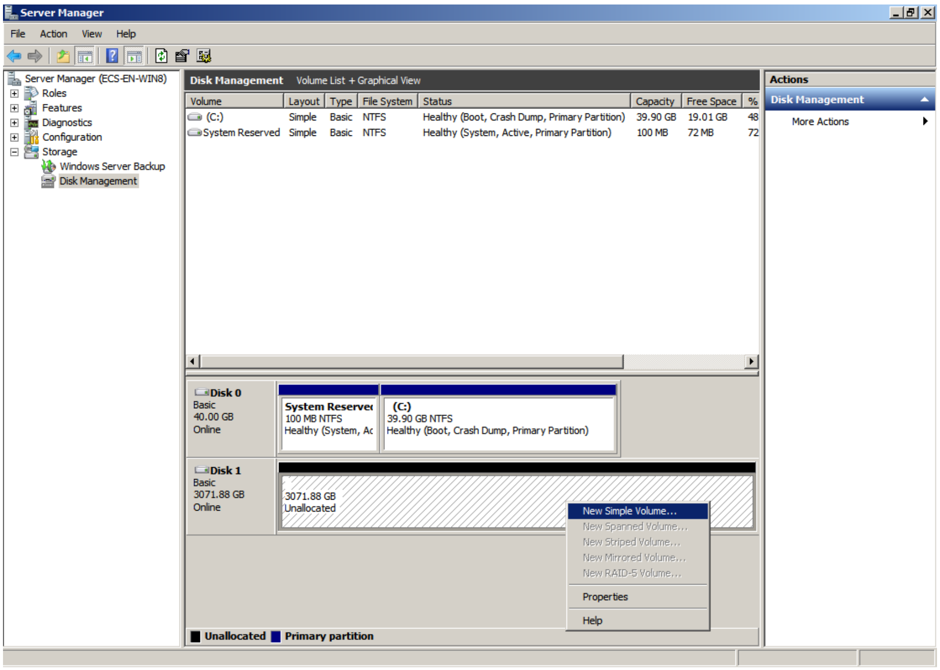
Figure 2-28 Initialize Disk (Windows Server 2008)



Step 5 In the **Initialize Disk** dialog box, the to-be-initialized disk is selected. In this example, the disk capacity is larger than 2 TiB. Therefore, select **GPT (GUID Partition Table)** and click **OK**.

The **Server Manager** window is displayed.

Figure 2-29 Server Manager (Windows Server 2008)



NOTICE

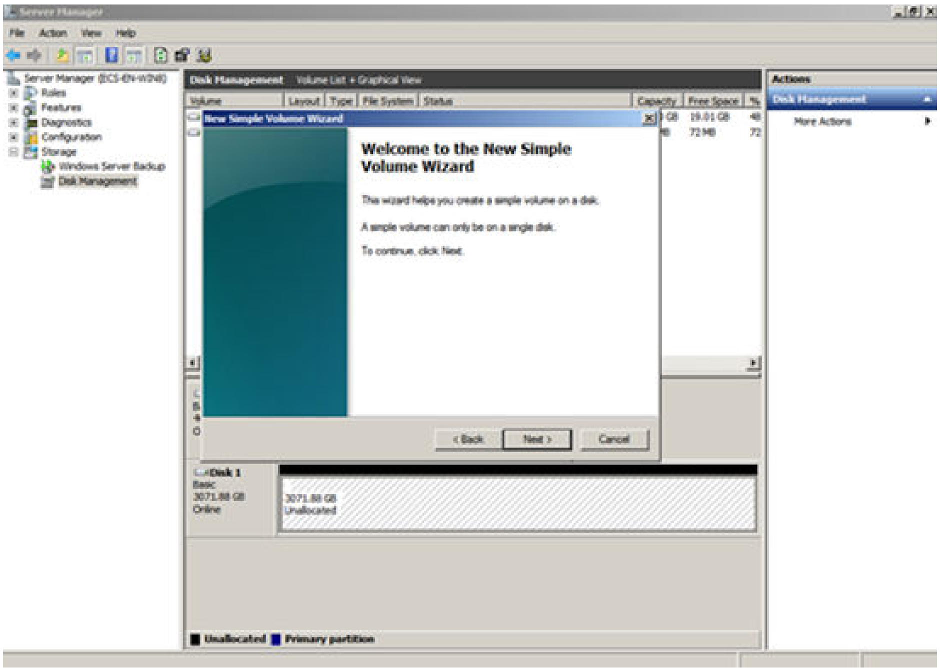
The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

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 6 Right-click at the unallocated disk space and choose **New Simple Volume** from the shortcut menu.

The **New Simple Volume Wizard** window is displayed.

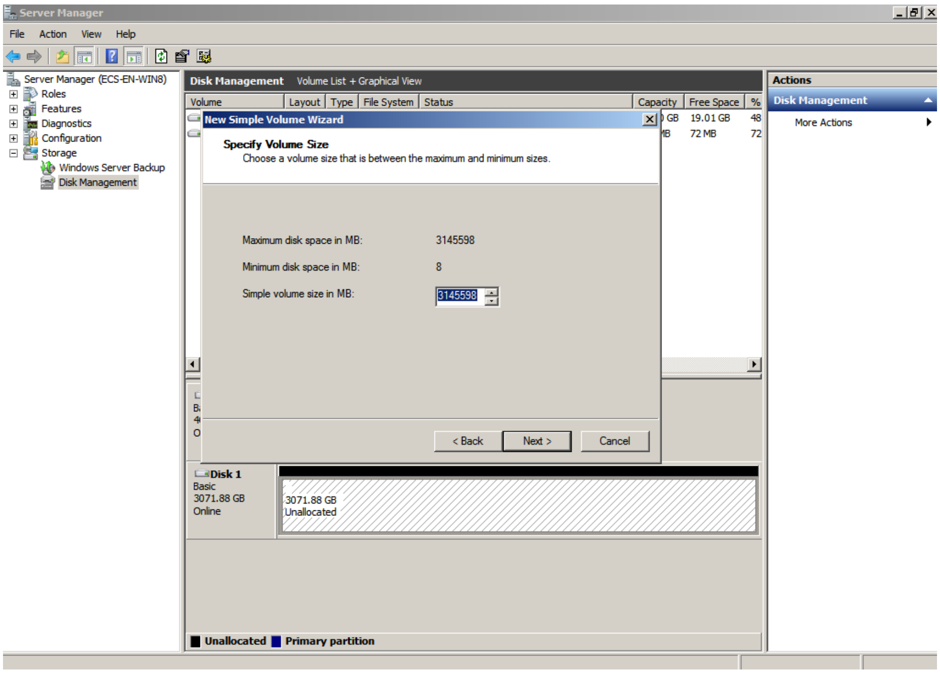
Figure 2-30 New Simple Volume Wizard (Windows Server 2008)



Step 7 Follow the prompts and click **Next**.

The **Specify Volume Size** page is displayed.

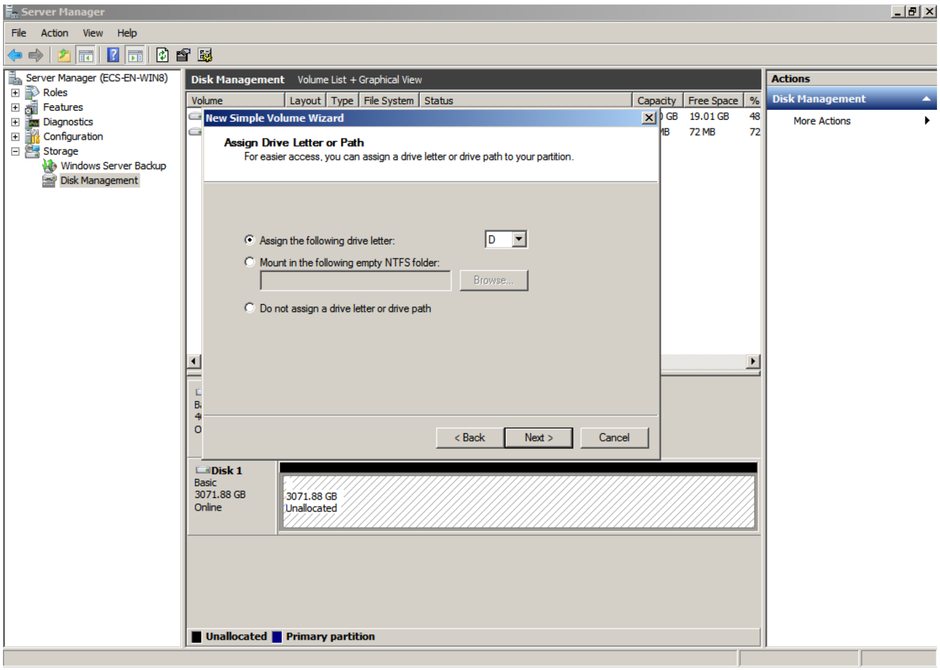
Figure 2-31 Specify Volume Size (Windows Server 2008)



Step 8 Specify the volume size and click **Next**. The system selects the maximum volume size by default. You can specify the volume size as required. In this example, the default setting is used.

The **Assign Drive Letter or Path** page is displayed.

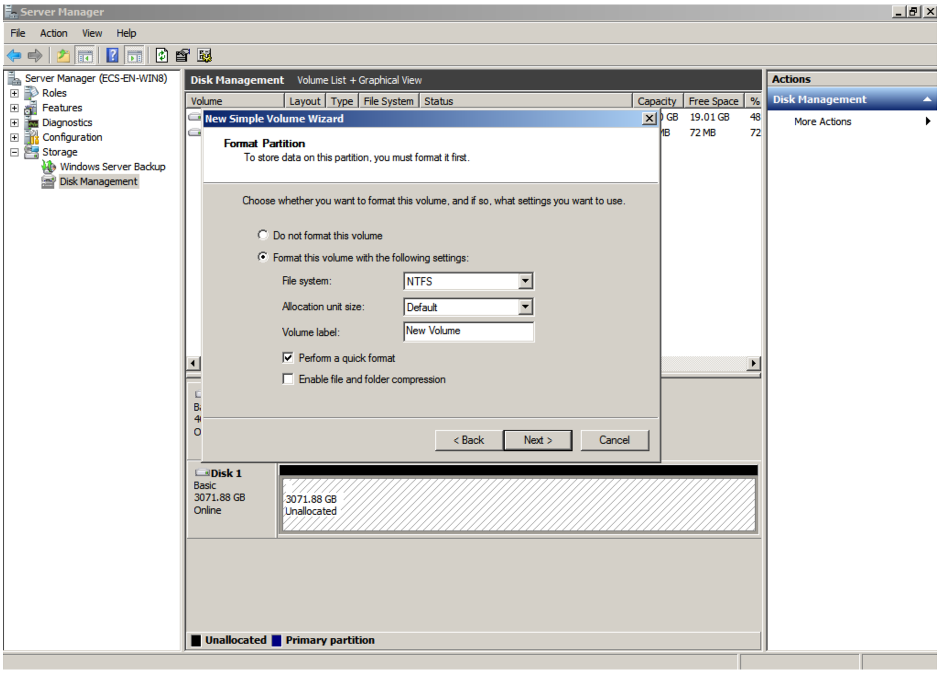
Figure 2-32 Assign Drive Letter or Path (Windows Server 2008)



Step 9 Assign a drive letter or path to your partition and click **Next**. The system assigns drive letter D by default. In this example, the default setting is used.

The **Format Partition** page is displayed.

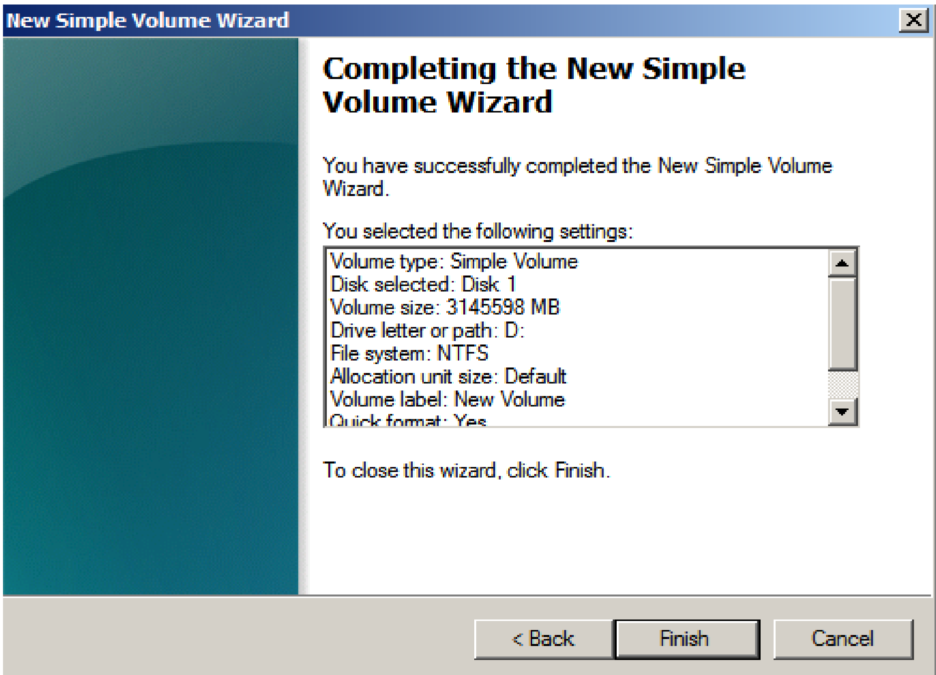
Figure 2-33 Format Partition (Windows Server 2008)



Step 10 Specify format settings and click **Next**. The system selects the NTFS file system by default. You can specify the file system type as required. In this example, the default setting is used.

The **Completing the New Simple Volume Wizard** page is displayed.

Figure 2-34 Completing the New Simple Volume Wizard



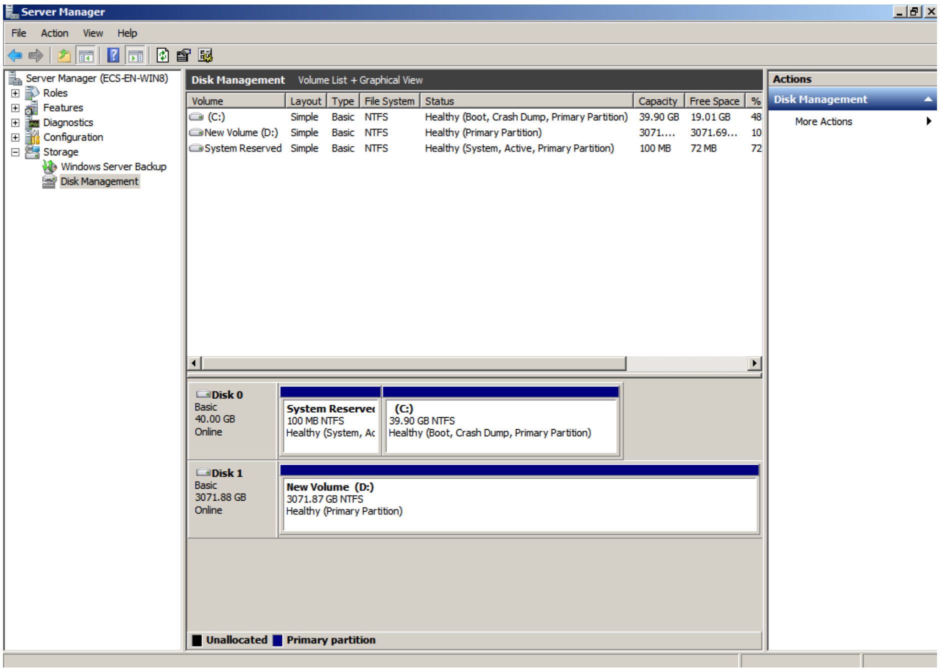
NOTICE

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

Step 11 Click **Finish**.

Wait for the initialization to complete. When the volume status changes to **Healthy**, the initialization has finished successfully.

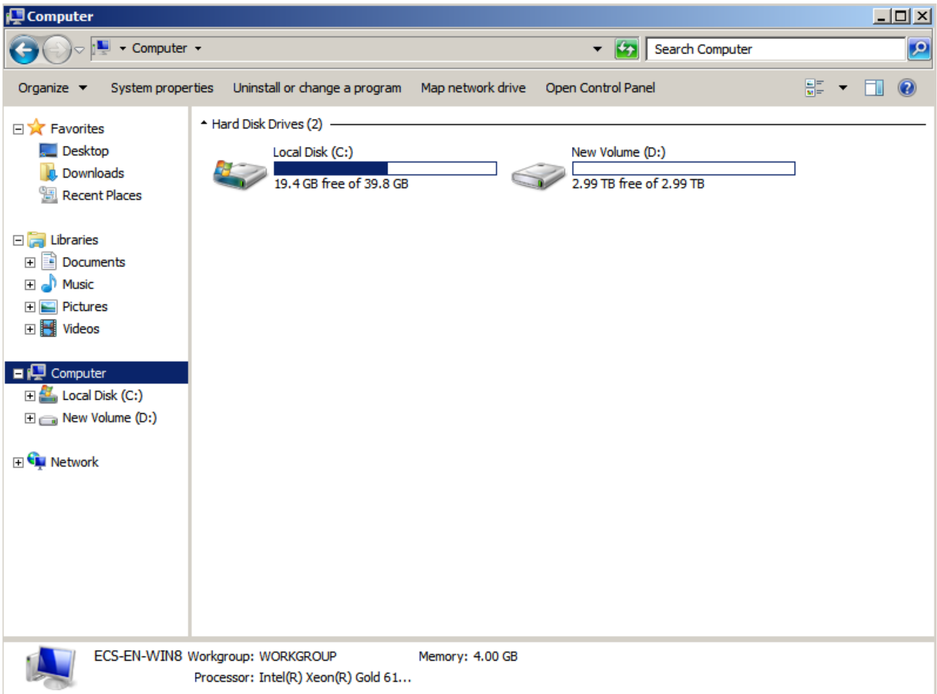
Figure 2-35 Disk initialization succeeded (Windows Server 2008)



Step 12 After the volume is created, click  and check whether a new volume appears in **Computer**. In this example, New Volume (D:) is the new volume.

If New Volume (D:) appears, the disk is successfully initialized and no further action is required.

Figure 2-36 Computer (Windows Server 2008)



----End

2.4.7 Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2012)

Scenarios

This section uses Windows Server 2012 R2 Standard 64bit to describe how 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 disk capacity supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Therefore, use the GPT partition style if your disk capacity is larger than 2 TiB. For details, see **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**. To learn more about disk partition styles, 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

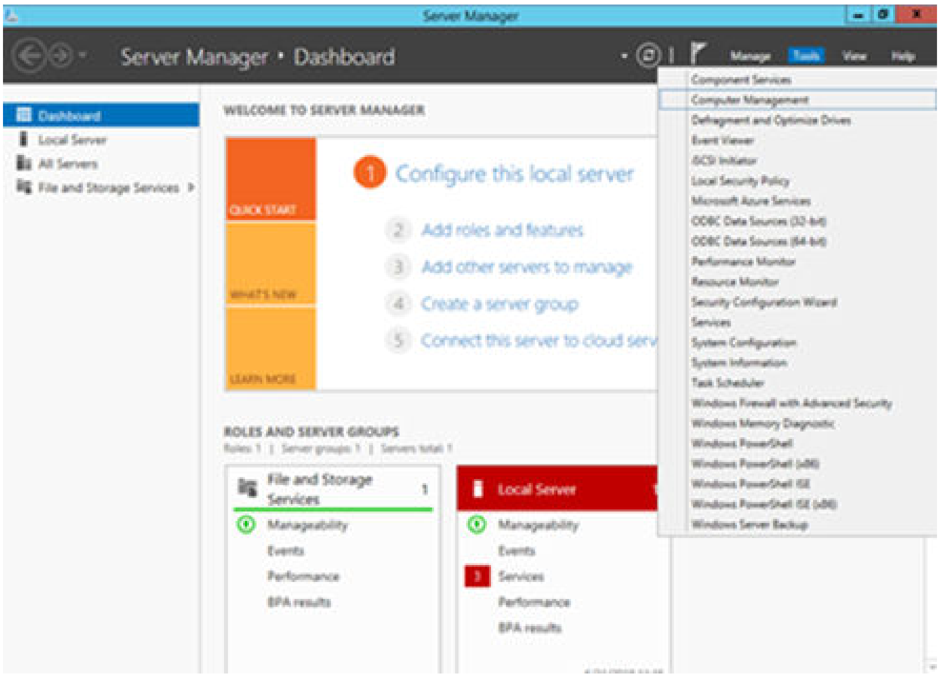
- You have attached a data disk to a server but not initialized the disk.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Procedure

Step 1 On the desktop of the server, click  in the lower area.

The **Server Manager** window is displayed.

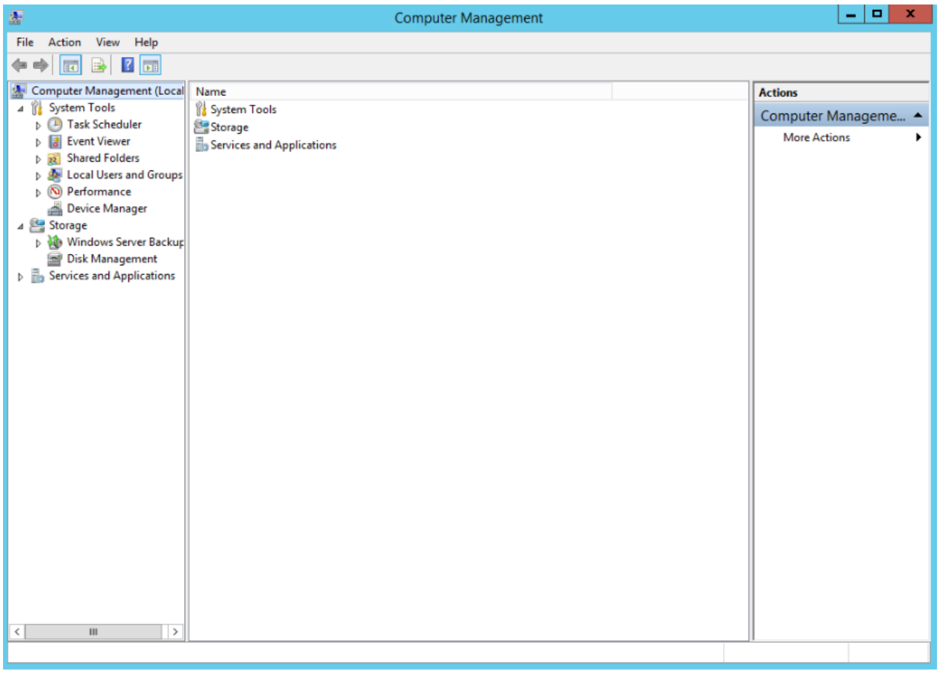
Figure 2-37 Server Manager (Windows Server 2012)



Step 2 In the upper right corner, choose **Tools** > **Computer Management**.

The **Computer Management** window is displayed.

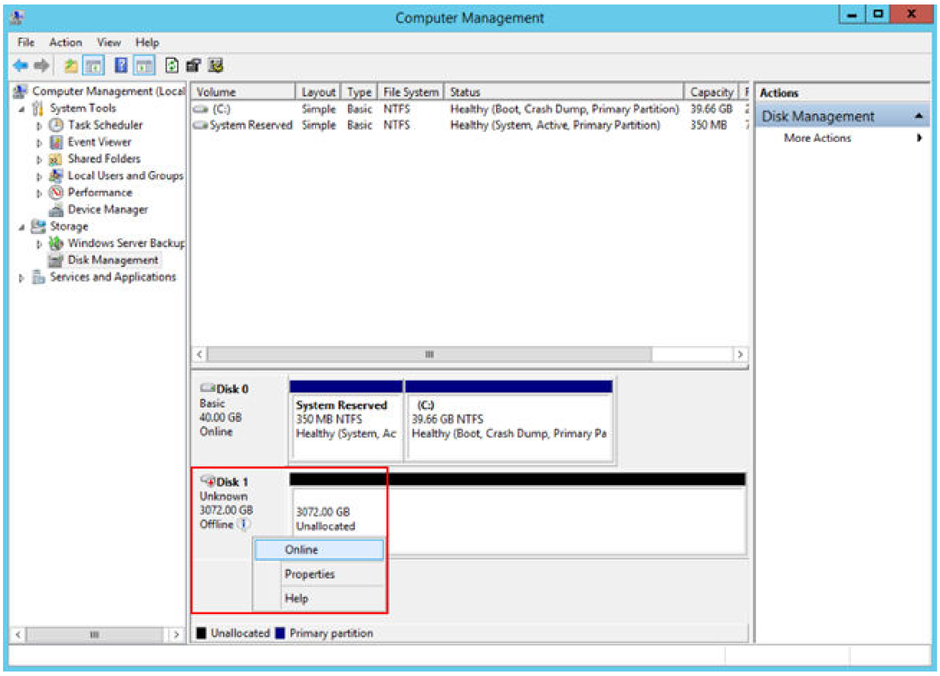
Figure 2-38 Computer Management window (Windows Server 2012)



Step 3 Choose **Storage** > **Disk Management**.

Disks are displayed in the right pane.

Figure 2-39 Disk Management list (Windows Server 2012)

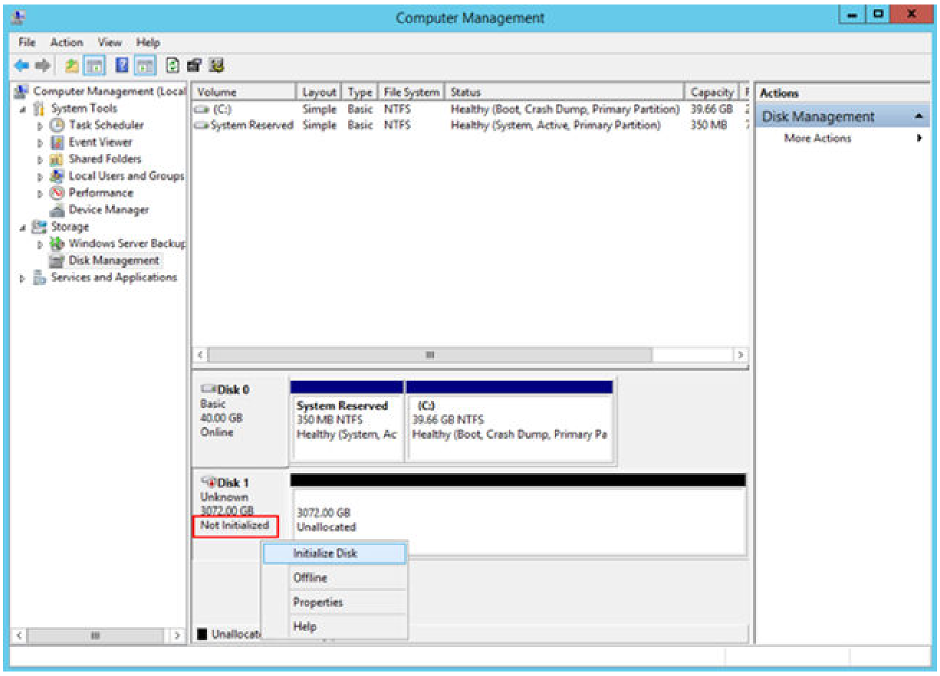


Step 4 (Optional) If the new disk is offline, bring it online before initializing it.

In the **Disk 1** area, right-click and choose **Online** from the shortcut menu.

When the status of Disk 1 changes from **Offline** to **Not Initialized**, the disk has been brought online.

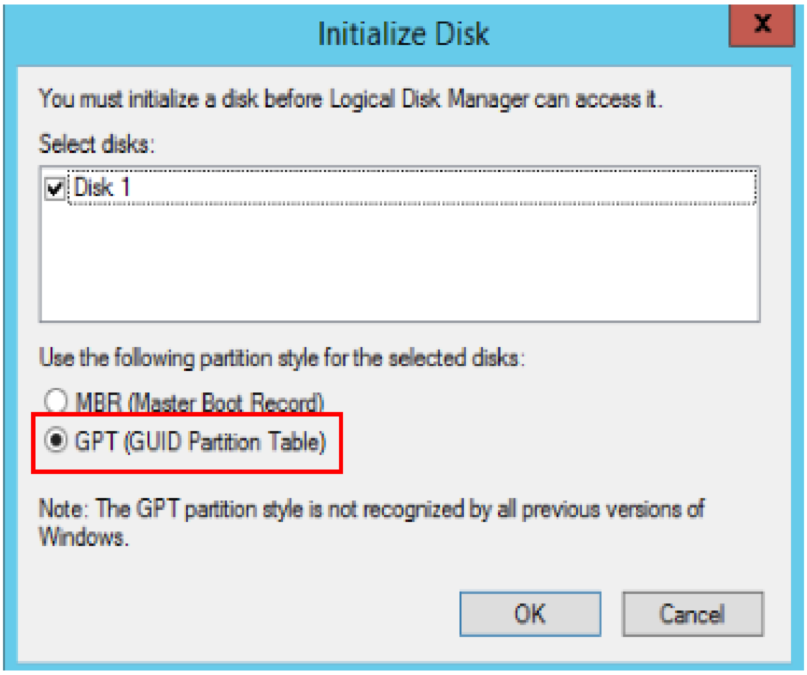
Figure 2-40 Bring online succeeded (Windows Server 2012)



Step 5 (Optional) In the **Disk 1** area, right-click and choose **Initialize Disk** from the shortcut menu.

The **Initialize Disk** dialog box is displayed.

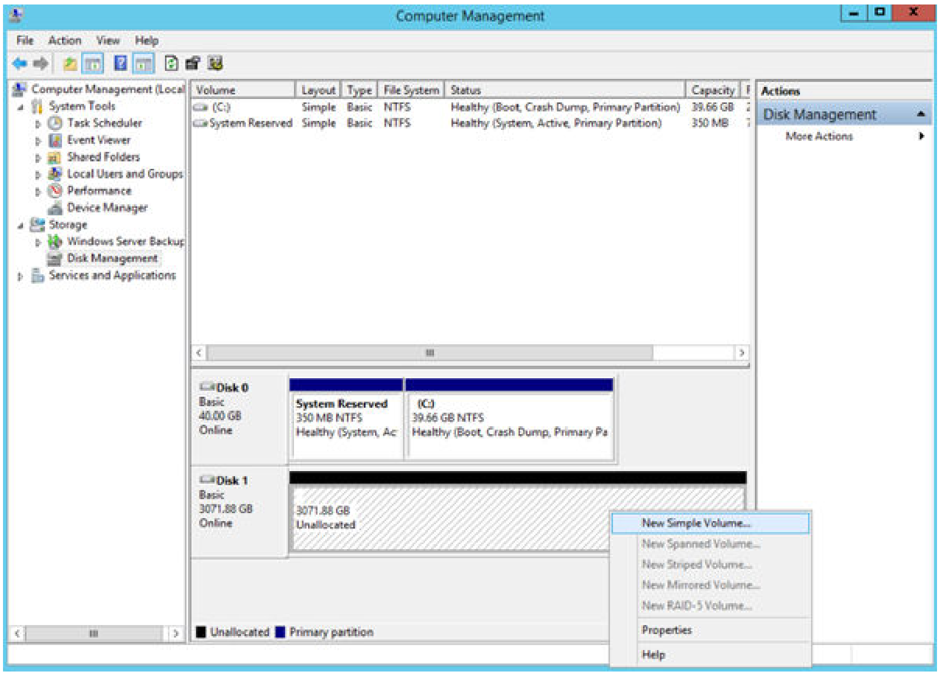
Figure 2-41 Initialize Disk (Windows Server 2012)



Step 6 In the **Initialize Disk** dialog box, the to-be-initialized disk is selected. In this example, the disk capacity is larger than 2 TiB. Therefore, select **GPT (GUID Partition Table)** and click **OK**.

The **Computer Management** window is displayed.

Figure 2-42 Computer Management (Windows Server 2012)



NOTICE

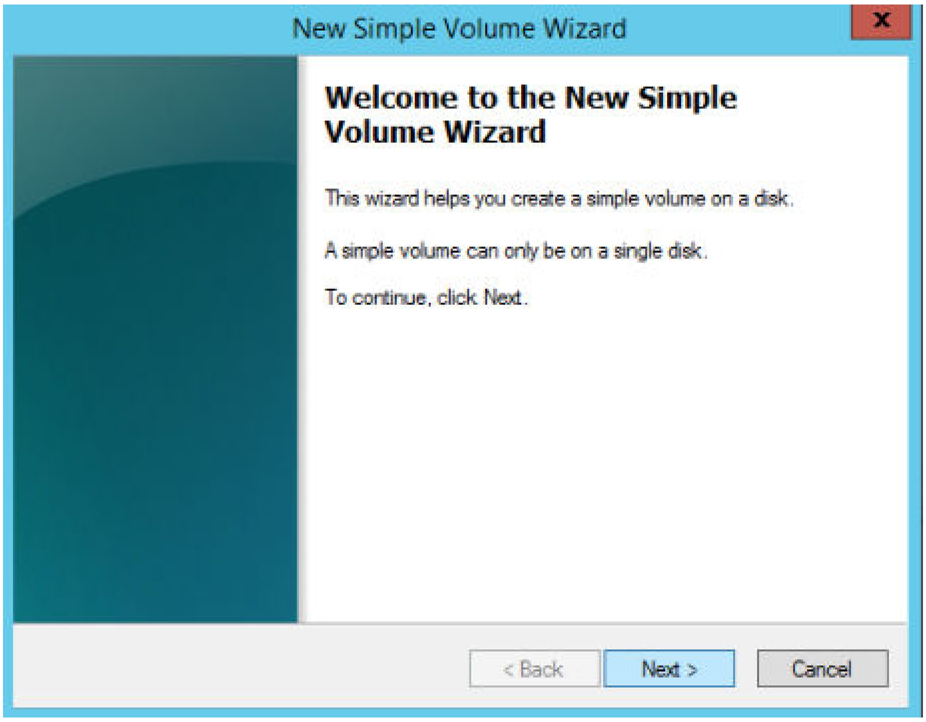
The maximum disk size supported by MBR is 2 TiB, and that supported by GPT is 18 EiB. Because an EVS data disk currently supports up to 32 TiB, use GPT if your disk size is larger than 2 TiB.

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 7 Right-click at the unallocated disk space and choose **New Simple Volume** from the shortcut menu.

The **New Simple Volume Wizard** window is displayed.

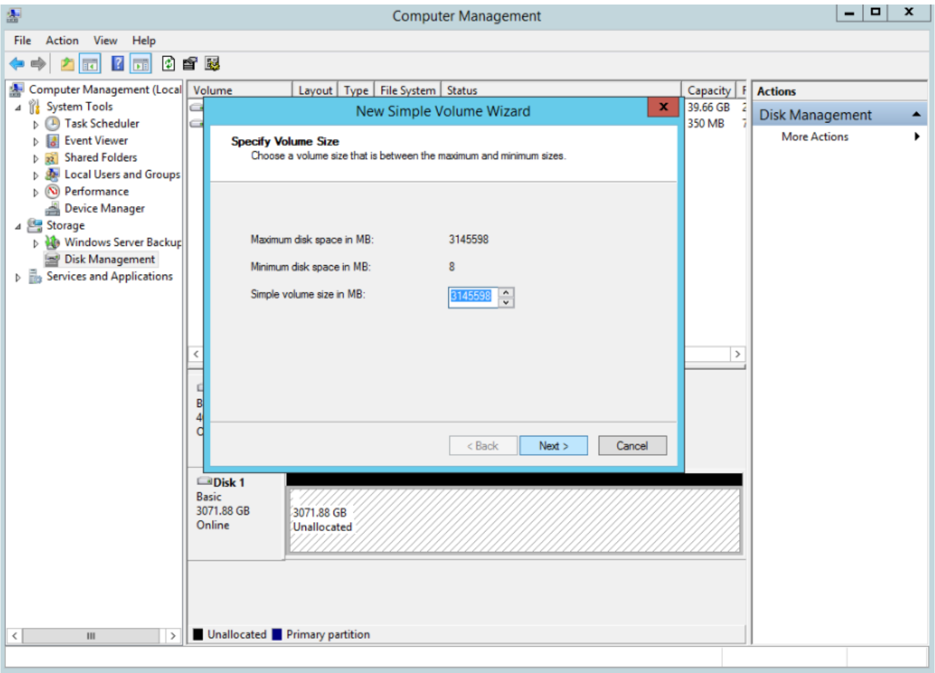
Figure 2-43 New Simple Volume Wizard (Windows Server 2012)



Step 8 Follow the prompts and click **Next**.

The **Specify Volume Size** page is displayed.

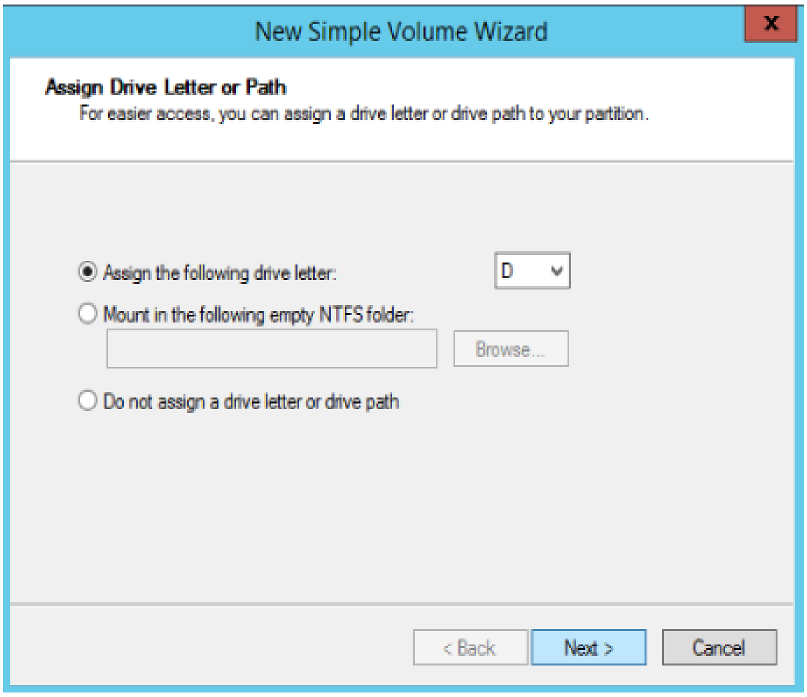
Figure 2-44 Specify Volume Size (Windows Server 2012)



Step 9 Specify the volume size and click **Next**. The system selects the maximum volume size by default. You can specify the volume size as required. In this example, the default setting is used.

The **Assign Drive Letter or Path** page is displayed.

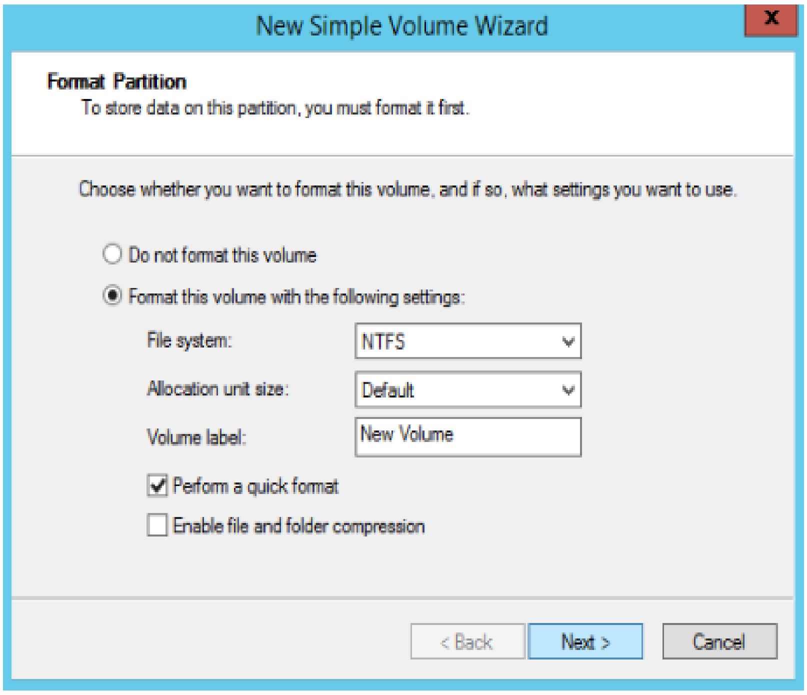
Figure 2-45 Assign Drive Letter or Path (Windows Server 2012)



Step 10 Assign a drive letter or path to your partition and click **Next**. The system assigns drive letter D by default. In this example, the default setting is used.

The **Format Partition** page is displayed.

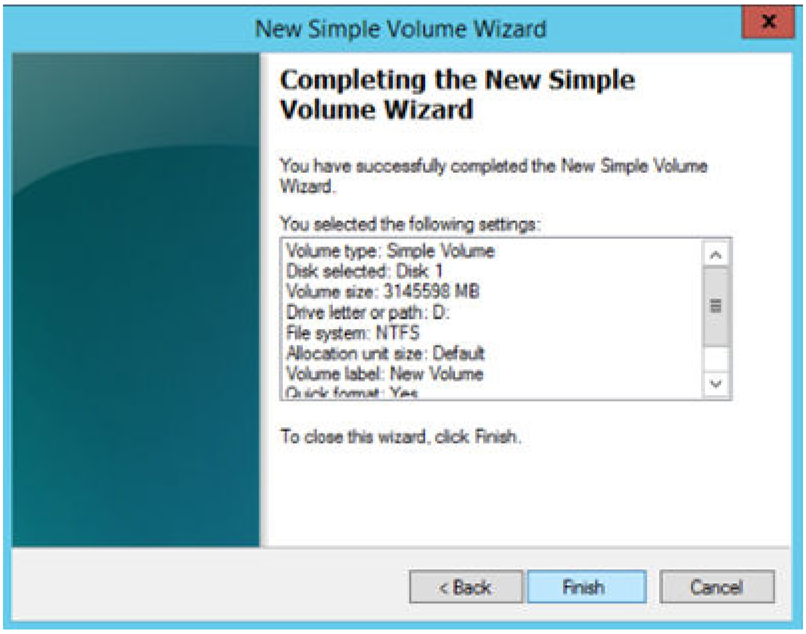
Figure 2-46 Format Partition (Windows Server 2012)



Step 11 Specify format settings and click **Next**. The system selects the NTFS file system by default. You can specify the file system type as required. In this example, the default setting is used.

The **Completing the New Simple Volume Wizard** page is displayed.

Figure 2-47 Completing the New Simple Volume Wizard (Windows Server 2012)



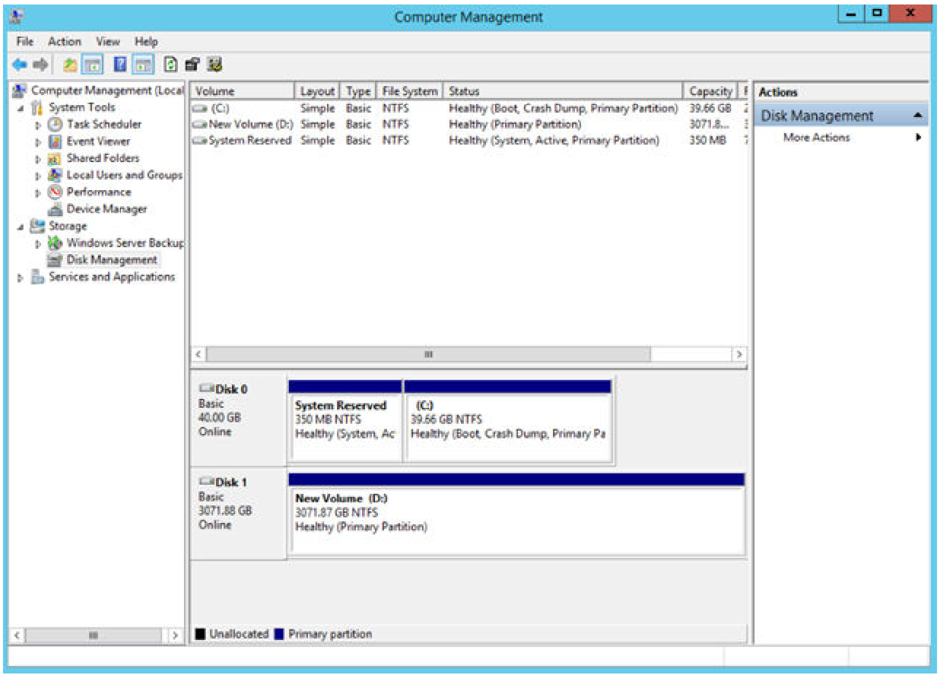
NOTICE

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

Step 12 Click **Finish**.

Wait for the initialization to complete. When the volume status changes to **Healthy**, the initialization has finished successfully.

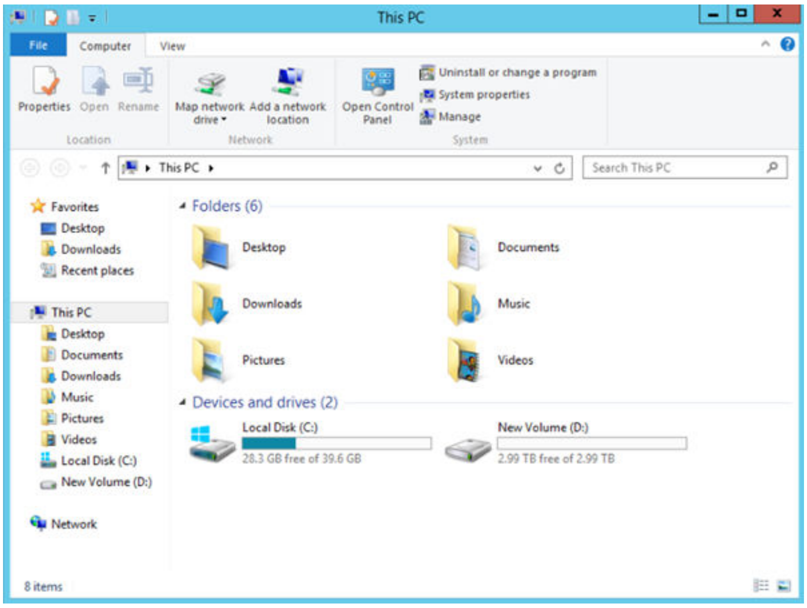
Figure 2-48 Disk initialization succeeded (Windows Server 2012)



Step 13 After the volume is created, click  and check whether a new volume appears in **This PC**. In this example, New Volume (D:) is the new volume.

If New Volume (D:) appears, the disk is successfully initialized and no further action is required.

Figure 2-49 This PC (Windows Server 2012)



----End

2.4.8 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

- You have attached a data disk to a server but not initialized the disk.

- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

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 server startup. You can configure the **fstab** file of a server that has data. This operation will not affect the existing data.

The following example uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names (like **/dev/vdb1**) 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 server stop or start. This can even prevent your server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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

3 Permissions Management

3.1 Creating a User and Granting EVS Permissions

This chapter describes how to use IAM to implement fine-grained permissions control for your EVS resources. With IAM, you can:

- Create IAM users for employees based on your enterprise's organizational structure. Each IAM user will have their own security credentials for accessing EVS resources.
- Grant only the permissions required for users to perform a specific task.
- Entrust an account or cloud service to perform professional and efficient O&M on your EVS resources.

If your account does not require individual IAM users, skip this chapter.

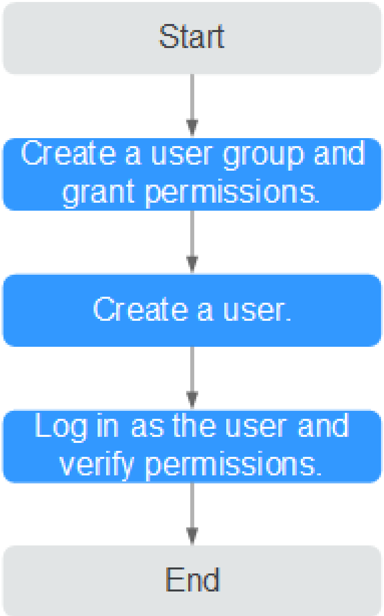
This section describes the procedure for granting permissions (see **Figure 3-1**).

Prerequisites

Learn about the permissions (see section "Permissions Management" in the *Elastic Volume Service User Guide*) supported by EVS and choose policies or roles according to your requirements.

Process Flow

Figure 3-1 Process for granting EVS permissions



1. Create a user group and assign permissions to it.

   Create a user group on the IAM console, and attach the **EVS ReadOnlyAccess** policy to the group.

2. Create a user and add it to a user group.

   Create a user on the IAM console and add the user to the group created in **1**.

3. Log in and verify permissions.

   Log in to the EVS console using the created user and verify that the user only has read permissions for EVS.

   – In **Service List**, choose **Elastic Volume Service**. On the EVS console, click **Create Disk** in the upper right corner. If a message appears indicating that you have insufficient permissions to perform the operation, the **EVS ReadOnlyAccess** policy has already taken effect.

   – Choose any other service in **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **EVS ReadOnlyAccess** policy has already taken effect.

3.2 EVS Custom Policies

Custom policies can be created to supplement the system-defined policies of EVS. For the actions supported for custom policies, see section "Permissions Policies and Supported Actions" in the *Elastic Volume Service API Reference*.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.

- JSON: Edit JSON policies from scratch or based on an existing policy.

  For operation details, see section "Creating a Custom Policy" in the *Identity and Access Management User Guide*. The following section contains examples of common EVS custom policies.

Example Custom Policies

- Example 1: Allowing users to create disks.

```
{
        "Version": "1.1",
        "Statement": [
                {
                        "Action": [
                                "evs:volumes:list",
                                "evs:volumes:get",
                                "evs:quotas:get",
                                "evs:volumeTags:list",
                                "evs:types:get",
                                "evs:volumes:create",
                                "ecs:cloudServerFlavors:get",
                                "ecs:cloudServers:list",
                                "bss:balance:view",
                                "bss:order:pay",
                                "bss:order:update"
                        ],
                        "Effect": "Allow"
                }
        ]
}
```

- Example 2: Denying disk deletion

  A policy with only "Deny" permissions must be used in conjunction with other policies to take effect. If the permissions assigned to a user contain both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

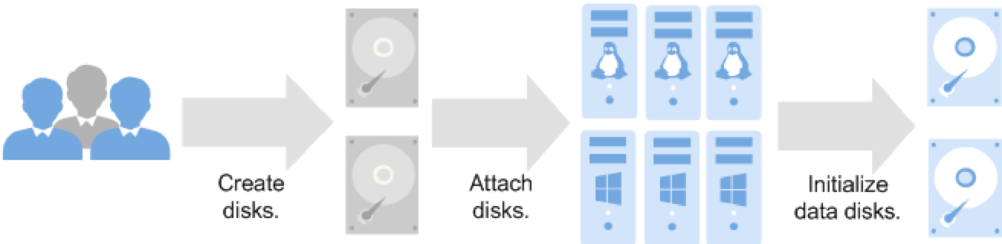
  The following method can be used if you need to assign permissions of the **EVS FullAccess** policy to a user but you want to prevent the user from deleting EVS disks. Create a custom policy for denying disk deletion, and attach both policies to the group to which the user belongs. Then, the user can perform all operations on disks except deleting disks. The following is an example of a deny policy:

```
{
        "Version": "1.1",
        "Statement": [
                {
                        "Effect": "Deny",
                        "Action": [
                                "evs:volumes:delete"
                        ]
                }
        ]
}
```

4 Creating and Using an EVS Disk

Figure 4-1 shows the basic EVS operation process.

Figure 4-1 Process overview



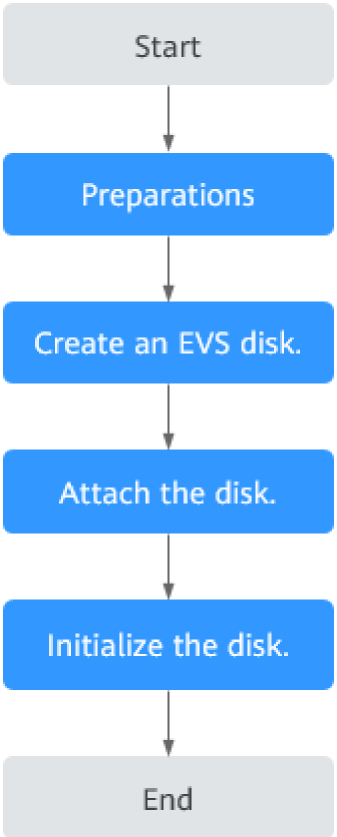
EVS disks can be attached to servers and be used as system disks or data disks. For details, see **Table 4-1**.

Table 4-1 Method of creation

Disk	Description	Method
System disk	System disks are created along with servers and cannot be created separately.	• See section "Creating an ECS" in the *Elastic Cloud Server User Guide*. • See section "Creating a BMS" in the *Bare Metal Server User Guide*.
Data disk	Data disks can be created along with servers or separately.	• See section "Creating an ECS" in the *Elastic Cloud Server User Guide*. • See section "Creating a BMS" in the *Bare Metal Server User Guide*. • **Create an EVS Disk**

Figure 4-2 shows how to purchase a data disk separately.

Figure 4-2 Process overview



1. **Preparations**: Register an account on the console and obtain the permissions required to create servers and disks.
2. **Create a disk.** Configure the disk parameters, including the disk type, size, name, and other information. For details, see **Create an EVS Disk**.
3. **Attach the data disk.** Attach the separately created 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**
   - Windows
     - **Initializing a Windows Data Disk (Windows Server 2008)**
     - **Initializing a Windows Data Disk (Windows Server 2019)**
     - **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2008)**
     - **Initializing a Windows Data Disk Larger Than 2 TiB (Windows Server 2012)**

– Linux

  - **Initializing a Linux Data Disk (fdisk)**
  - **Initializing a Linux Data Disk (parted)**
  - **Initializing a Linux Data Disk Larger Than 2 TiB (parted)**

5 Disk Capacity Expansion

5.1 Expansion Overview

What Is Capacity Expansion?

If the capacity of an existing disk is insufficient, you can expand the disk capacity to increase the storage space.

Both system disks and data disks can be expanded. Disk capacity can only be expanded, not reduced.

NOTE

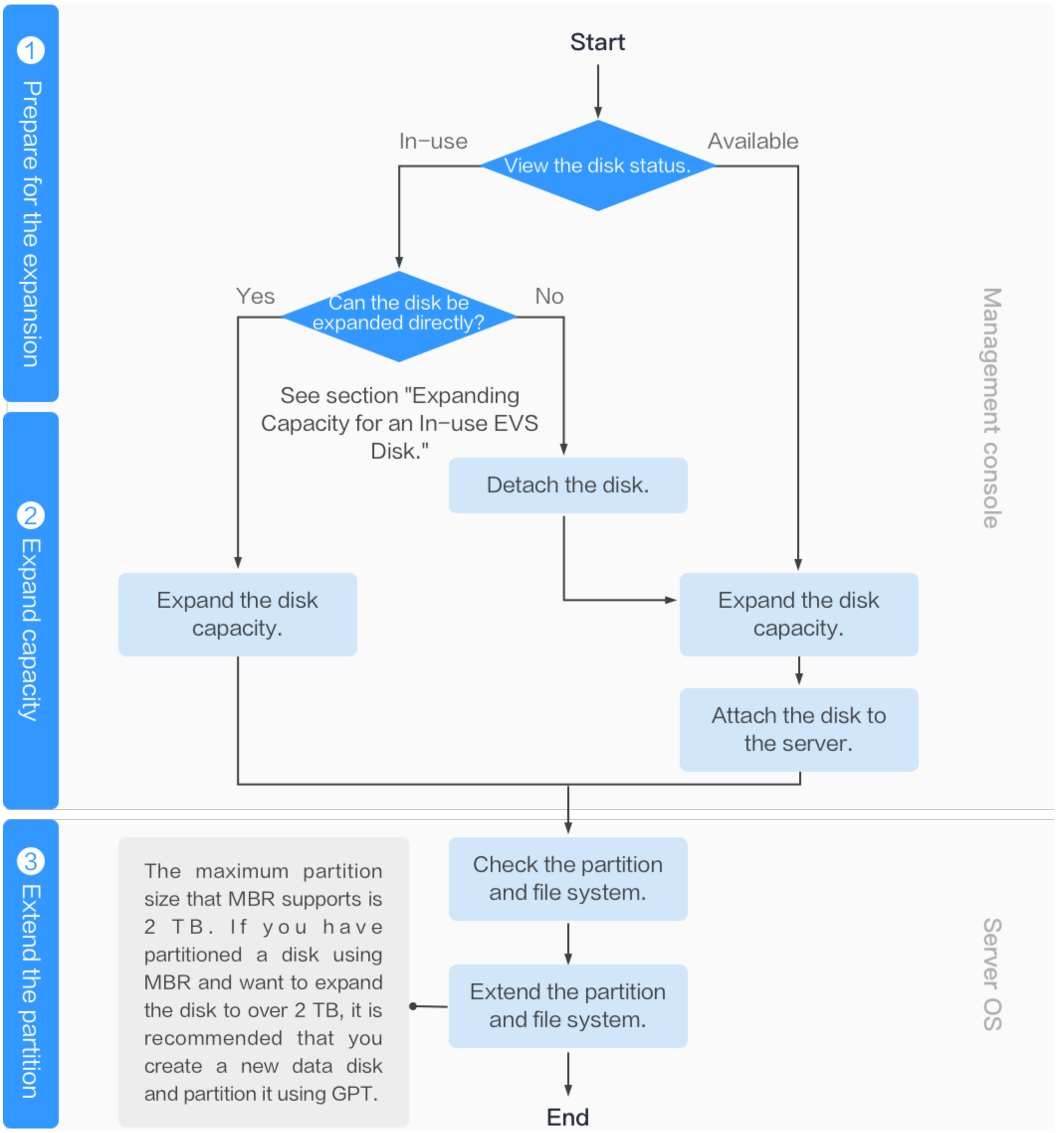
If you detach a system disk and then attach it to an ECS as a data disk, the maximum capacity of this disk is still 1 TiB.

How to Expand the Disk Capacity?

A capacity expansion operation includes the following steps:

1. **Expand the disk capacity on the management console.**
2. **Log in to the server and extend the disk partition and file system.**

Figure 5-1 Capacity expansion procedure



Expand the Disk Capacity on the Management Console

Choose a proper expansion method based on the disk status.

- For an In-use disk:

  The disk has been attached to a server. Check whether the disk can be expanded in the In-use state by referring to **Constraints**.

  – If yes, expand the disk capacity according to **Expanding Capacity for an In-use EVS Disk**.

  – If no, detach the disk. Then, expand the disk capacity according to **Expanding Capacity for an Available EVS Disk**.

- For an Available disk:

  The disk has not been attached to any server and can be directly expanded by referring to **Expanding Capacity for an Available EVS Disk**.

  A shared disk can be expanded only when its status is **Available**.

Log In to the Server and Extend the Disk Partition and File System

After the disk has been expanded on the management console, only the disk storage capacity is enlarged, but its additional space cannot be used directly. You must log in to the server and extend the disk partition and file system. For details, see **Table 5-1**.

Table 5-1 Extending the disk partition and file system

Capacity After Expansion	Extend Disk Partition and File System
Disk capacity ≤ 2 TiB	• Windows: **Extending Disk Partitions and File Systems (Windows Server 2008)** • Linux: **Partition and File System Extension Preparations (Linux)**
Disk capacity > 2 TiB	• GPT partition style: **Extending Disk Partitions and File Systems (Windows Server 2008)** or **Partition and File System Extension Preparations (Linux)** • MBR partition style: Not supported The maximum disk capacity that MBR supports is 2 TiB, and the disk space exceeding 2 TiB cannot be used. If your disk uses MBR and you need to expand the disk capacity to over 2 TiB, change the partition style from MBR to GPT. Ensure that the disk data has been backed up before changing the partition style because services will be interrupted and data on the disk will be deleted during this change.

NOTE

If the server is stopped during the expansion, the additional space of a Windows system disk, Windows data disk, or Linux system disk may be automatically added to the last partition after the server is started. In this case, the additional space can be directly used. If the additional space is not automatically added, extend the partition and file system according to the preceding steps.

Related Operations

For more information, see **FAQs** > **Capacity Expansion**.

5.2 Expanding Capacity for an In-use EVS Disk

Scenarios

This section describes how to expand the capacity of an In-use EVS disk on the management console. The In-use status indicates that the disk has been attached to a server. You do not need to detach the disk when expanding an In-use disk.

Constraints

- Disk capacity can be expanded, but cannot be reduced.
- When expanding an In-use disk, the server attached with this disk must be in the **Running** or **Stopped** state.
- A shared disk in the **In-use** state cannot be expanded. To expand such a disk, you must detach it from all its servers, wait until its status changes to **Available**, and then expand its capacity. For more information, see **Expanding Capacity for an Available EVS Disk**.
- Only some server OSs support capacity expansion of In-use disks. For servers without such support, detach the disk and then expand its capacity. Otherwise, you may need to stop and then start the server after the expansion to make the additional space available.

  Perform the following operations to check whether your server OS supports capacity expansion of In-use disks:

  a. Check your server image. Both public images and private images listed on the console support the capacity expansion of In-use disks.

     How to view: Log in to the management console. In the navigation pane on the left, click ☰ and choose **Compute** > **Image Management Service**. On the **Public Images** tab, view the images of the **ECS image** type.

  b. If your server OS does not appear in the image list, check whether it is included in **Table 5-2**.

     If yes, you can expand the disk. Otherwise, you must detach the disk and then expand its capacity. For details, see **Expanding Capacity for an Available EVS Disk**.

Table 5-2 OSs that support the capacity expansion of In-use disks

OS	Version
CentOS 8	8.0 64-bit or later
CentOS 7	7.2 64-bit or later
CentOS 6	6.5 64-bit or later
Debian	8.5.0 64-bit or later
Fedora	24 64-bit or later

OS	Version
SUSE 12	SUSE Linux Enterprise Server 12 64-bit or later
SUSE 11	SUSE Linux Enterprise Server 11 SP4 64-bit
OpenSUSE	42.1 64-bit or later
Oracle Linux Server release 7	7.2 64-bit or later
Oracle Linux Server release 6	6.7 64-bit or later
Ubuntu Server	14.04 64-bit or later
Red Hat Enterprise Linux 7	7.3 64-bit
Red Hat Enterprise Linux 6	6.8 64-bit
EulerOS	2.2 64-bit or later
Windows Server 2016	Windows Server 2016 R2 Enterprise 64-bit
Windows Server 2012	Windows Server 2012 R2 Standard 64-bit
Windows Server 2008	Windows Server 2008 R2 Enterprise 64-bit

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 Choose a way to expand the disk by determining whether you want to check server information first.

- If yes, perform the following procedure:
  a. In the disk list, click the name of the to-be-expanded disk.
     The disk details page is displayed.
  b. Click the **Attachments** tab to view the server where the target disk has been attached.
  c. Click **Expand Capacity** in the upper right corner of the page.
     The expansion page is displayed.
- If no, perform the following procedure:
  a. In the disk list, locate the row that contains the target disk and choose **More** > **Expand Capacity** in the **Operation** column.
     The expansion page is displayed.

Step 4 Set the **New Capacity** parameter and click **Next**.

Step 5 On the **Details** page, check the disk details.

- Click **Submit** to start the expansion.
- Click **Previous** to change the settings.

After the configuration is submitted, go back to the disk list page.

Step 6 In the disk list, view the capacity of the target disk.

When the disk status changes from **Expanding** to **In-use** and the disk capacity increases, the expansion has succeeded.

NOTE

When the status of a disk is **Expanding**, you are not allowed to modify the specifications of the ECS where the disk is attached.

NOTE

If the expansion fails, technical support personnel will contact you and help you handle this error. Do not perform any operations on the disk before the technical support personnel contact you. If you require that the error be handled as soon as possible, contact our technical support personnel. Disks whose capacities failed to be expanded are not billed.

Step 7 Log in to the server and extend the partition and file system after the disk has been expanded on the console, because previous steps only enlarge the disk space.

The operations vary depending on the server OS.

- In Windows, see **Extending Disk Partitions and File Systems (Windows Server 2008)**.
- In Linux, see **Partition and File System Extension Preparations (Linux)**.

----End

5.3 Expanding Capacity for an Available EVS Disk

Scenarios

This section describes how to expand the capacity of an Available EVS disk on the management console. The Available status indicates that the disk has not been attached to any server.

Constraints

- Disk capacity can be expanded, but cannot be reduced.
- A shared disk in the **In-use** state cannot be expanded. To expand such a disk, you must detach it from all its servers, wait until its status changes to **Available**, and then expand its capacity.

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 In the disk list, locate the row that contains the target disk and choose **More** > **Expand Capacity** in the **Operation** column.

The expansion page is displayed.

Step 4 Set the **New Capacity** parameter and click **Next**.

Step 5 On the **Details** page, check the disk details.

- Click **Submit** to start the expansion.
- Click **Previous** to change the settings.

After the configuration is submitted, go back to the disk list page.

Step 6 In the disk list, view the capacity of the target disk.

When the disk status changes from **Expanding** to **Available** and the disk capacity increases, the expansion has succeeded.

NOTE

When the status of a disk is **Expanding**, you are not allowed to modify the specifications of the ECS where the disk is attached.

NOTE

If the expansion fails, technical support personnel will contact you and help you handle this error. Do not perform any operations on the disk before the technical support personnel contact you. If you require that the error be handled as soon as possible, contact our technical support personnel. Disks whose capacities failed to be expanded are not billed.

Step 7 Attach the disk to the server. For details, see **Attach an EVS Disk**.

Step 8 Log in to the server and extend the partition and file system after the disk has been expanded on the console, because previous steps only enlarge the disk space.

The operations vary depending on the server OS.

- In Windows, see **Extending Disk Partitions and File Systems (Windows Server 2008)**.
- In Linux, see **Partition and File System Extension Preparations (Linux)**.

----End

5.4 Extending Disk Partitions and File Systems (Windows Server 2008)

Scenarios

After a disk is expanded on the management console, the disk size is enlarged, but the additional space cannot be used directly.

In Windows, you must allocate the additional space to an existing partition or a new partition.

If the disk capacity is expanded on a stopped server, the additional space of a Windows system disk or Windows data disk will be automatically added to the partition at the end of the disk upon the server startup. In this case, the additional space can be used directly.

This section uses Windows Server 2008 R2 Enterprise 64bit as the sample OS to describe the expansion methods:

- For a system disk:
  - If volume (C:) already exists, you can add the additional space to volume (C:) and use it as a system volume. For details, see **System Disk: Add the Additional Space to the Original Volume**.
  - If volume (C:) already exists, you can create a new volume such as volume (F:) with the additional space and use the new volume as a data volume. For details, see **System Disk: Create a New Volume with the Additional Space**.
  - If the additional space has been added to volume (C:), you can shrink volume (C:), create a new volume with the available space, and use the new volume as a data volume. Only the available space can be shrunk and used to create the new volume. The additional space cannot be shrunk if it has already been used. This section uses a system disk to describe how to perform extension operations for a Windows disk. These operations are also suitable for data disks. For details, see **System Disk: Create a New Volume Using the Available Space Shrunk from the Original Volume**.
- For a data disk:
  - If volume (D:) already exists, you can add the additional space to volume (D:) and use it as a data volume. For details, see **Data Disk: Add the Additional Space to the Original Volume**.
  - If volume (D:) already exists, you can create a new volume such as volume (E:) with the additional space and use the new volume as a data volume. For details, see **Data Disk: Create a New Volume with the Additional Space**.

The method for allocating the additional space varies with the server OS. This section is used for reference only. For detailed operations and differences, see the corresponding OS documents.

NOTICE

Performing the expansion operations with caution. Incorrect operations may lead to data loss or exceptions, so you are advised to back up the disk data using CBR or snapshots before expansion. For details about using CBR, see **Managing EVS Backups**. For details about using snapshots, see the *Elastic Volume Service API Reference*

NOTICE

Prerequisites

- You have expanded the disk capacity and attached the disk to a server on the management console. For details, see **Expanding Capacity for an In-use EVS Disk** or **Expanding Capacity for an Available EVS Disk**.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

System Disk: Add the Additional Space to the Original Volume

In this example, the system disk has 50 GiB originally, and 22 GiB is added on the management console. The following procedure describes how to add this 22 GiB to volume (C:) on the server. After the operation is complete, volume (C:) will have 72 GiB of capacity and can be used as a system volume.

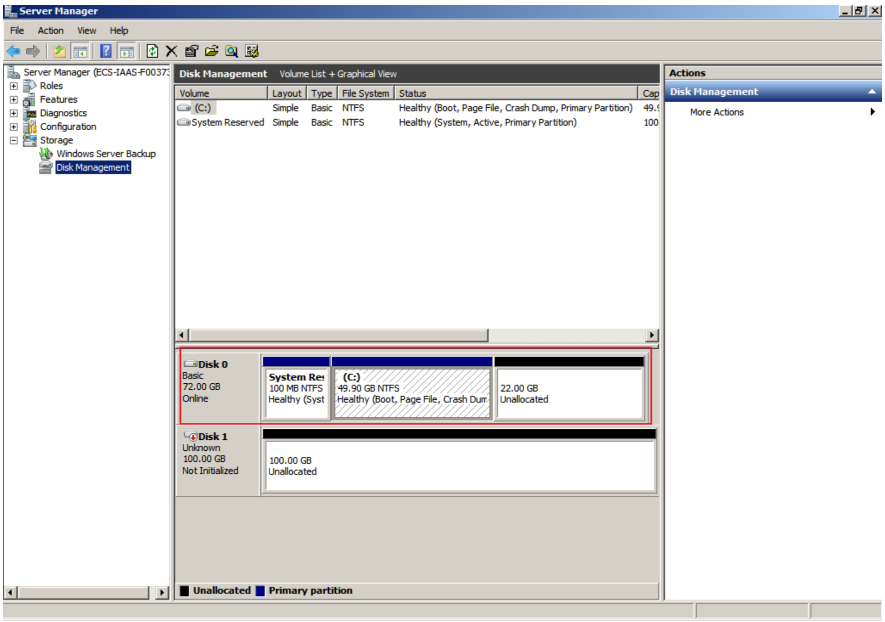
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-2 Disk Management (system disk)

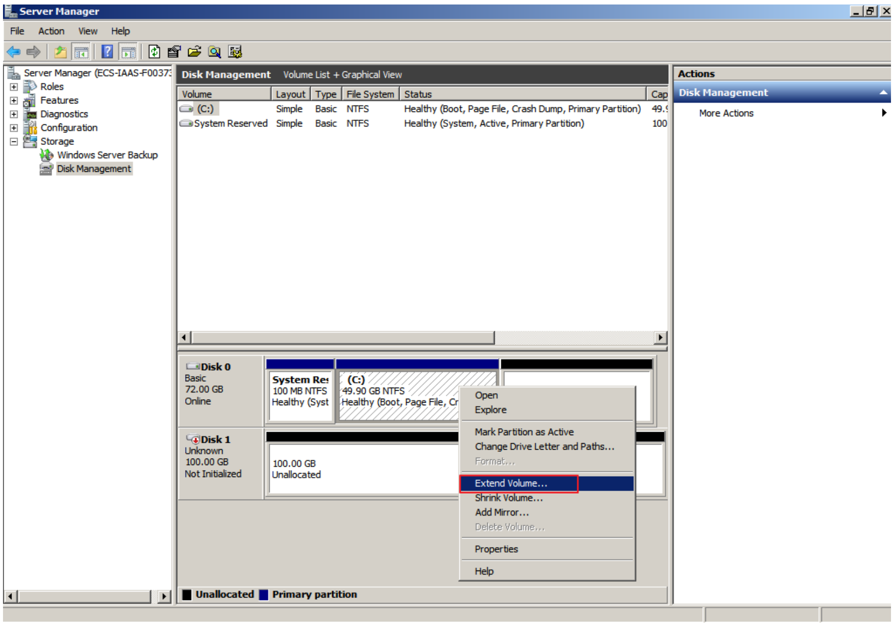


> **NOTE**
>
> If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

Step 3 On the **Disk Management** page, select the disk and volume that you want to extend. The current volume size and unallocated space are displayed.

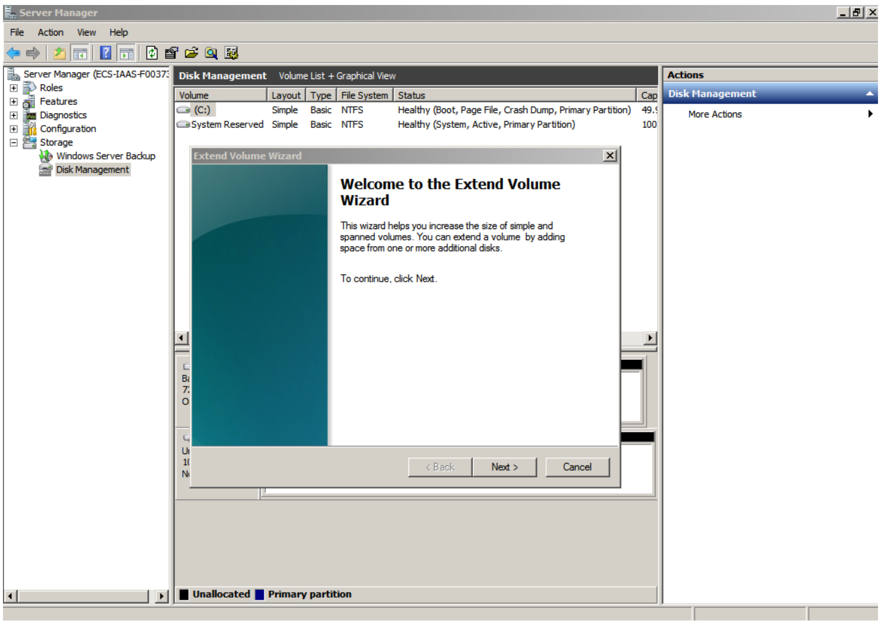
Step 4 Right-click the target volume and choose **Extend Volume**.

Figure 5-3 Choosing **Extend Volume**



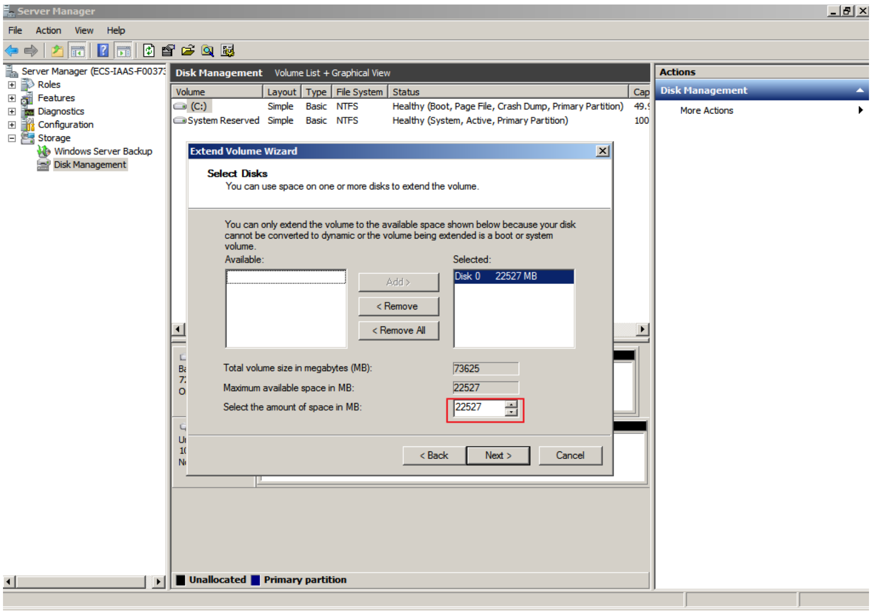
Step 5 On the displayed **Extend Volume Wizard** window, click **Next**.

Figure 5-4 Extend Volume Wizard



Step 6 In the text box to the right of **Select the amount of space in MB**, enter the amount of the additional space and click **Next**.

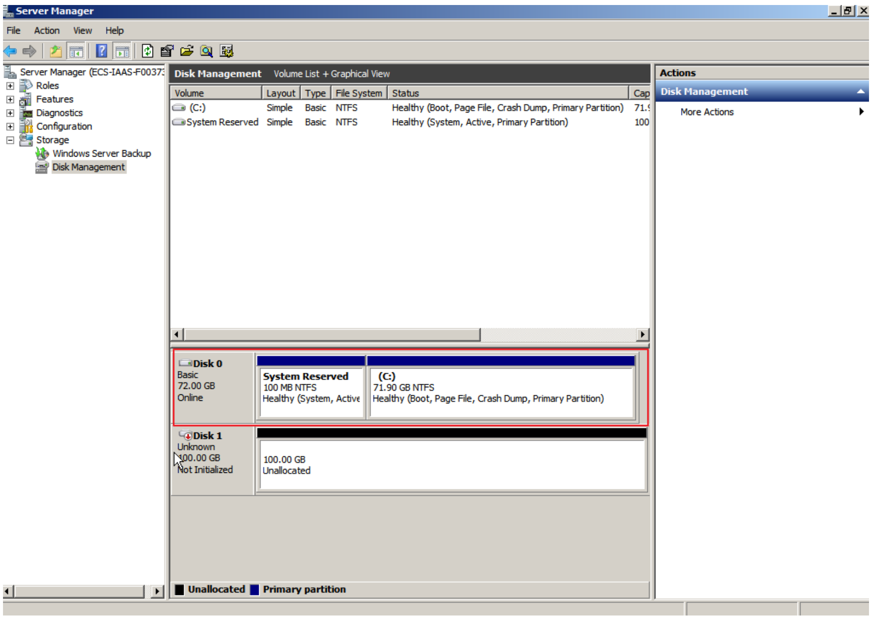
Figure 5-5 Selecting space



Step 7 Click **Finish**.

After the expansion succeeded, the partition size is larger than the original size.

Figure 5-6 Capacity expansion succeeded



----End

System Disk: Create a New Volume with the Additional Space

In this example, the system disk has 40 GiB originally, and 60 GiB is added on the management console. The following procedure describes how to use this 60 GiB to create a new volume, for example volume (F:), on the server. After the operation is complete, new volume (F:) has 60 GiB of capacity and can be used as a data volume.

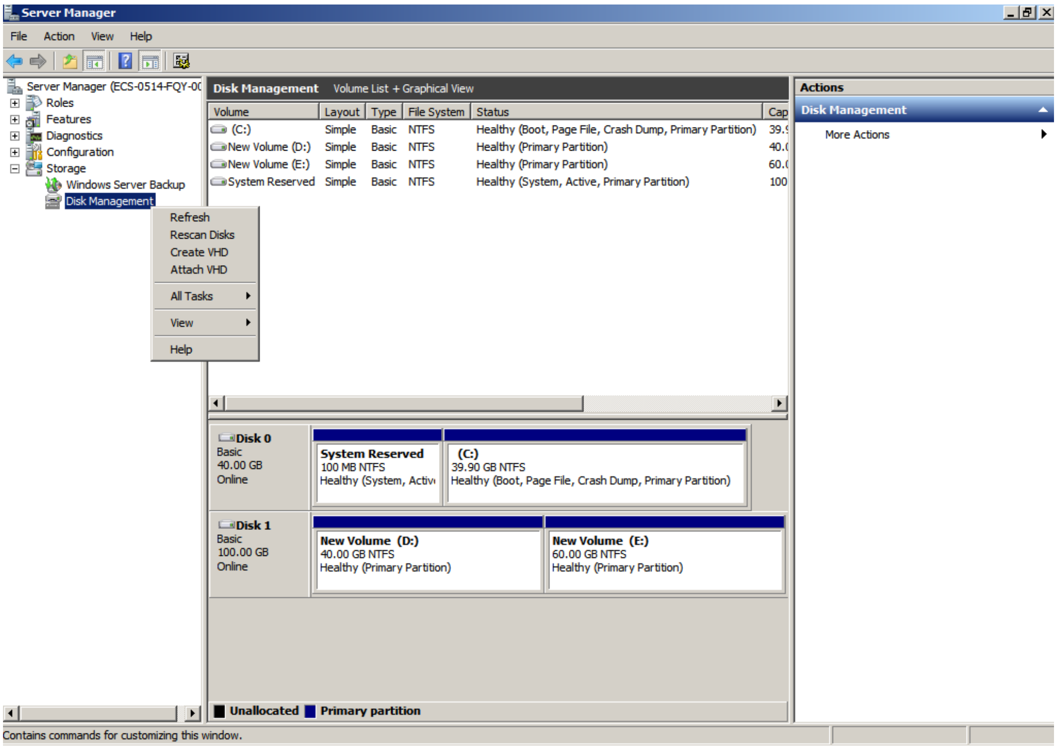
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The **Disk Management** window is displayed.

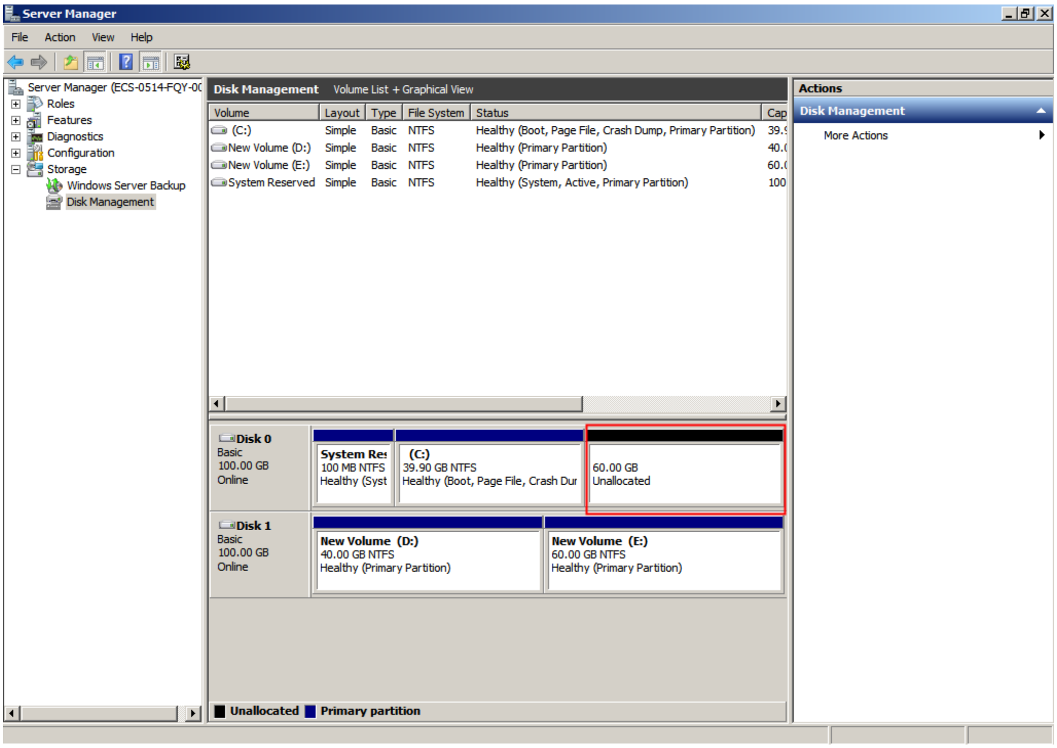
Figure 5-7 Refresh (system disk)



Step 3 If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

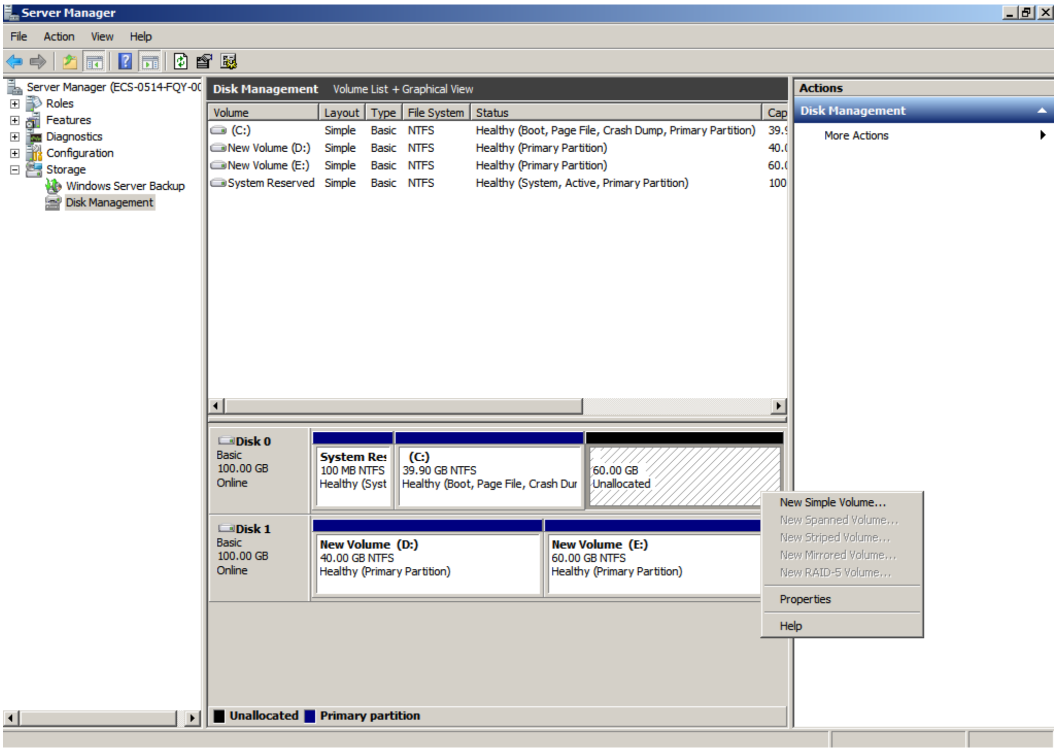
After the refresh, the additional space is displayed in the right area and is unallocated.

Figure 5-8 Unallocated disk space (system disk)



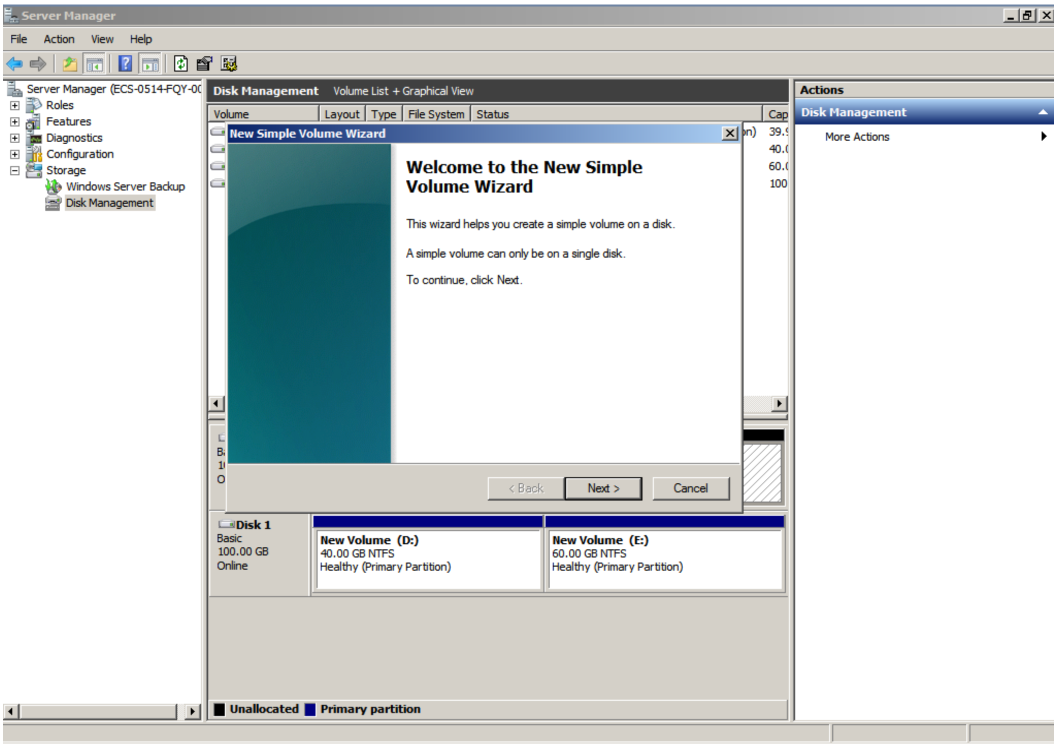
Step 4 In the **Unallocated** area of **Disk 0**, right-click the blank area and choose **New Simple Volume**.

Figure 5-9 New Simple Volume (system disk)



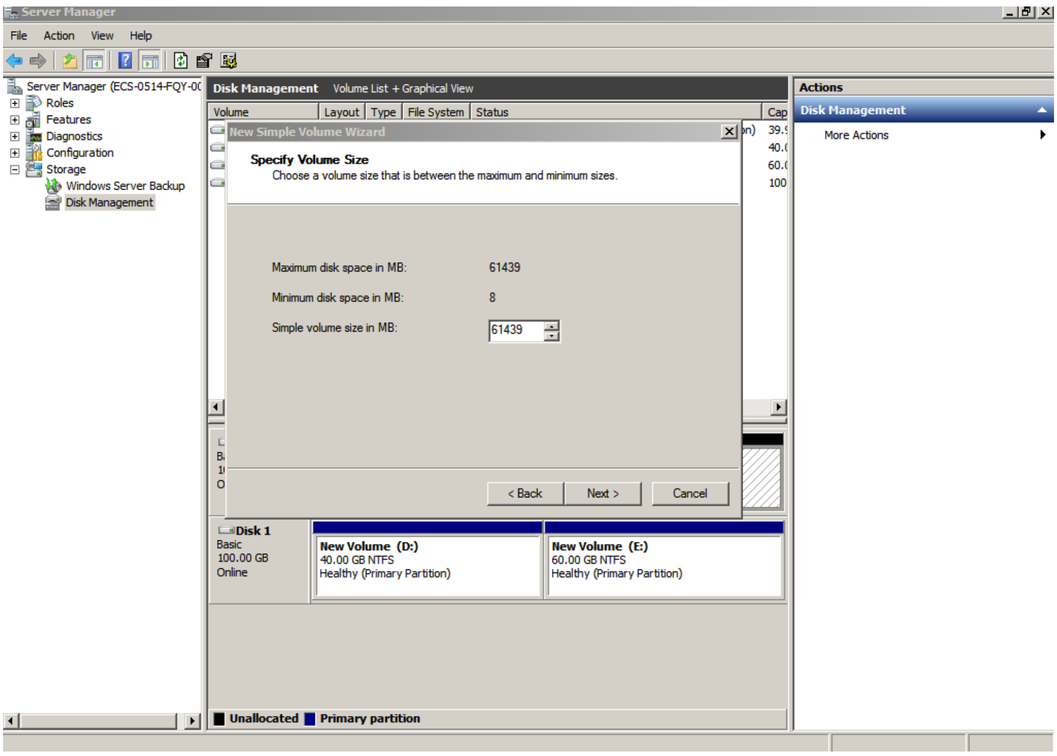
Step 5 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 5-10 New Simple Volume Wizard (system disk)



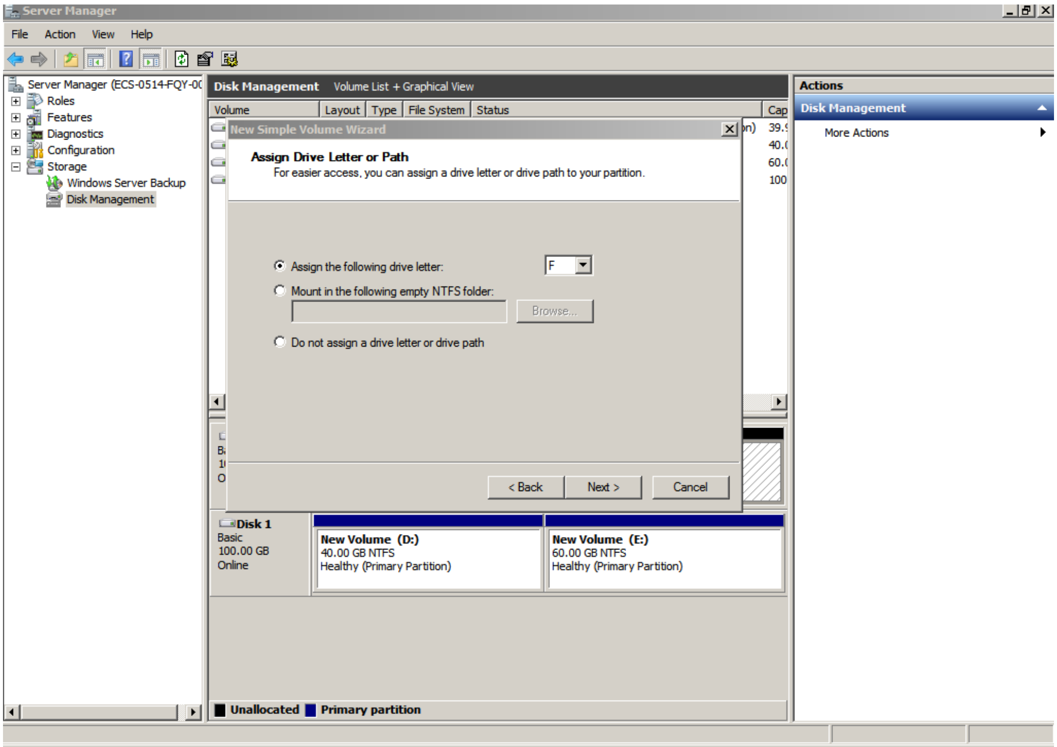
Step 6 On the displayed **Specify Volume Size** page, set **Simple volume size in MB** and click **Next**. In this example, the default size is used.

Figure 5-11 Specify Volume Size (system disk)



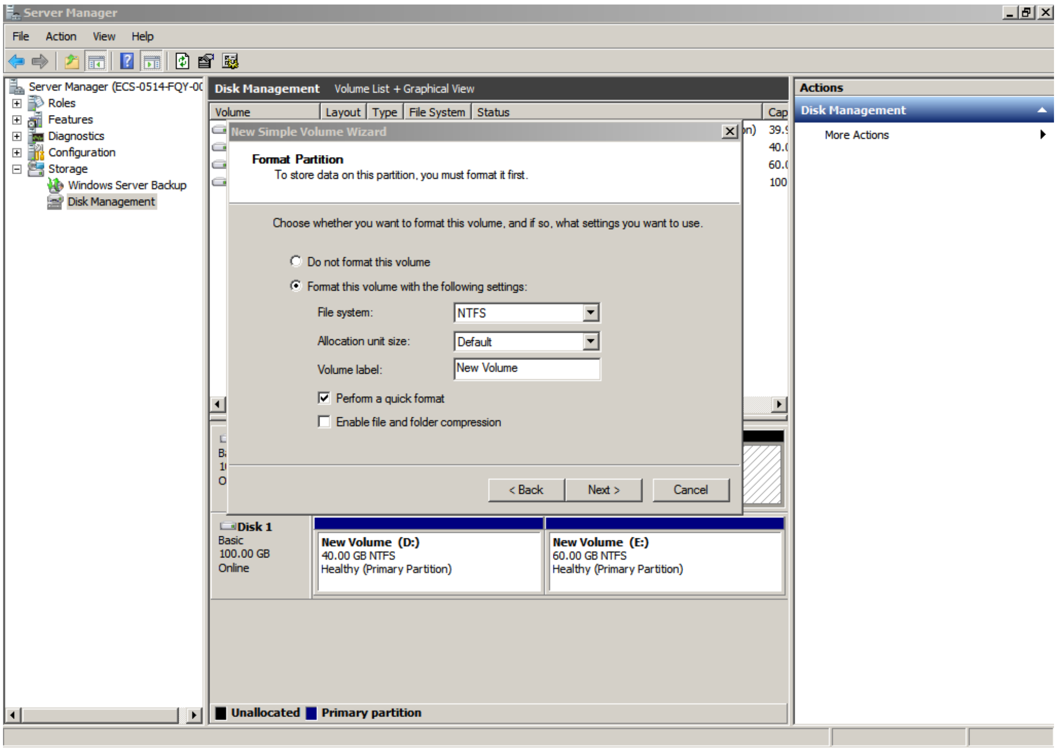
Step 7 On the displayed **Assign Drive Letter and Path** page, click **Assign the following drive letter**, select a drive letter, and click **Next**. In this example, drive letter **F** is selected.

Figure 5-12 Assign Drive Letter or Path (system disk)



Step 8 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

Figure 5-13 Format Partition (system disk)



Step 9 Click **Finish**.

After the expansion succeeded, new volume (F:) is displayed.

Figure 5-14 Completing the New Simple Volume Wizard (new volume F:)

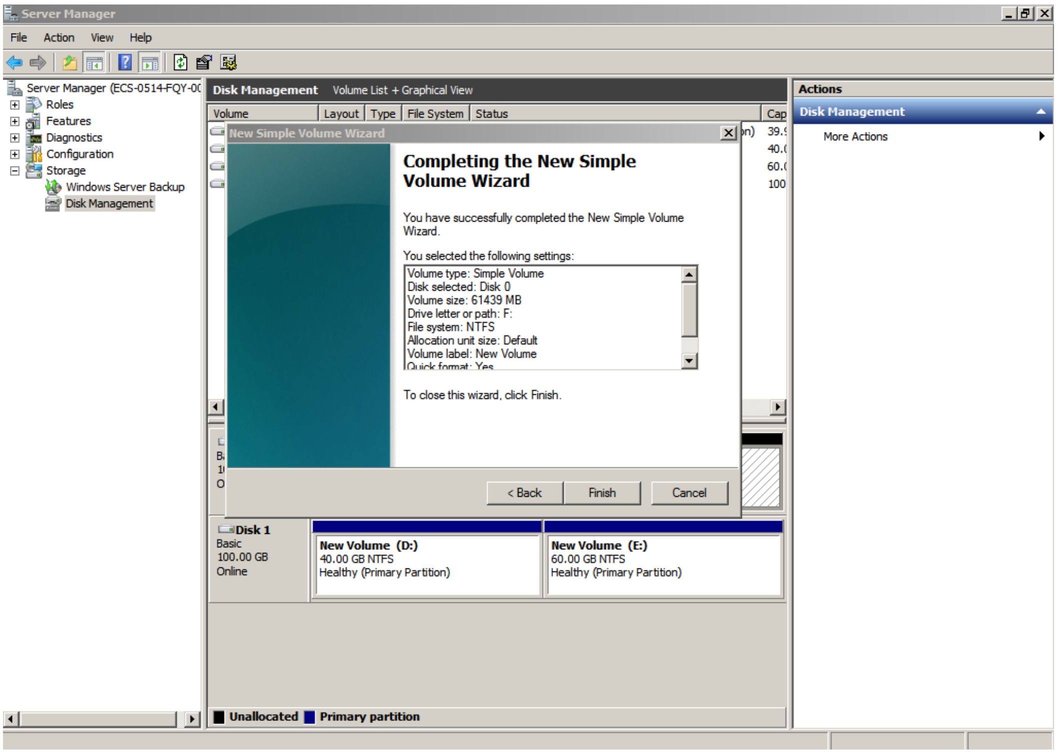
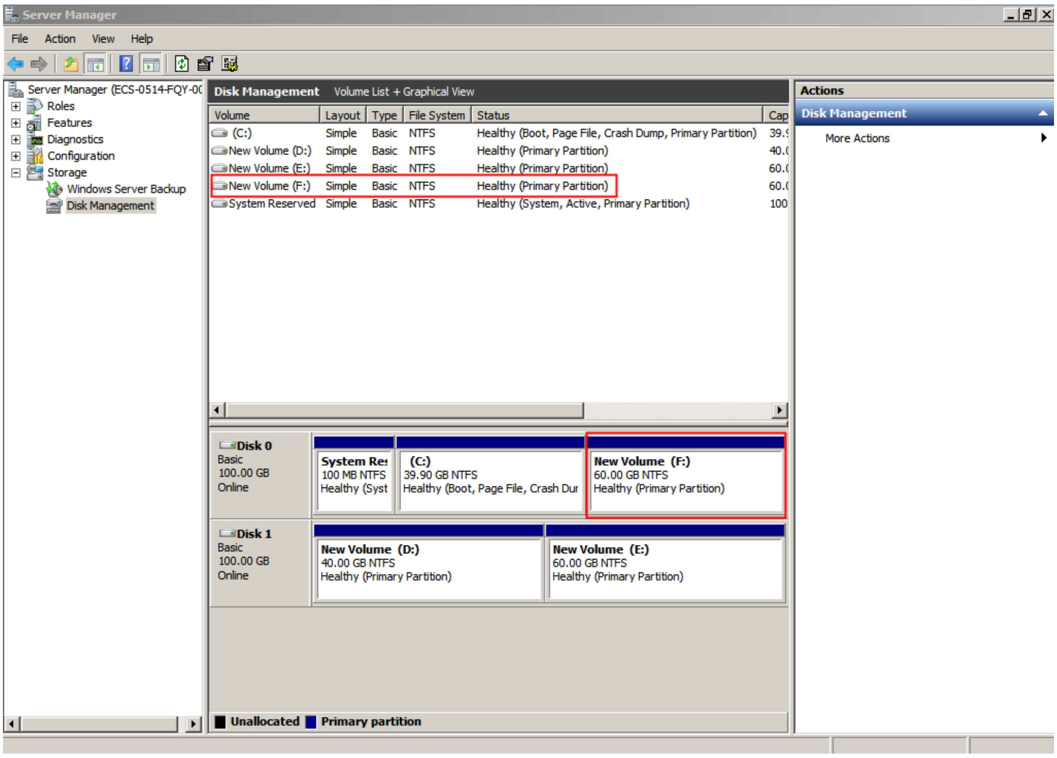


Figure 5-15 New Volume (F:)



----End

System Disk: Create a New Volume Using the Available Space Shrunk from the Original Volume

In this example, the system disk has 40 GiB originally, and 60 GiB is added on the management console and then formatted and added to volume (C:). This 60 GiB has not been used.

The following procedure describes how to use the shrink function to create new volume (D:) with this 60 GiB. After the operation is complete, new volume (D:) can be used as a data volume.

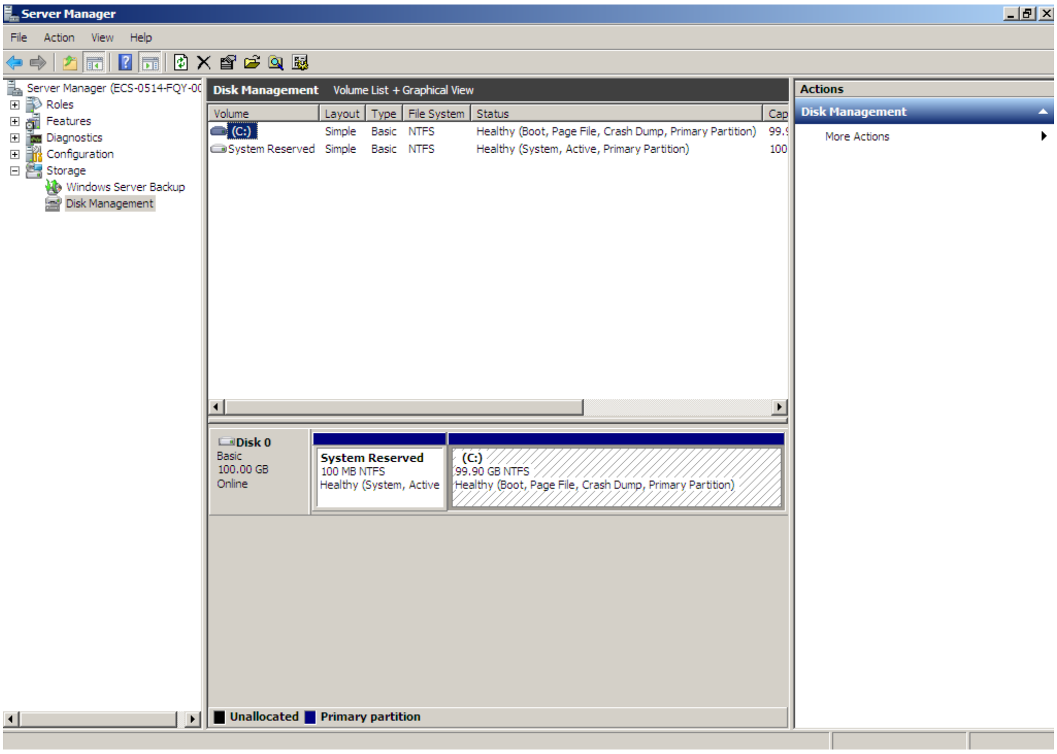
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

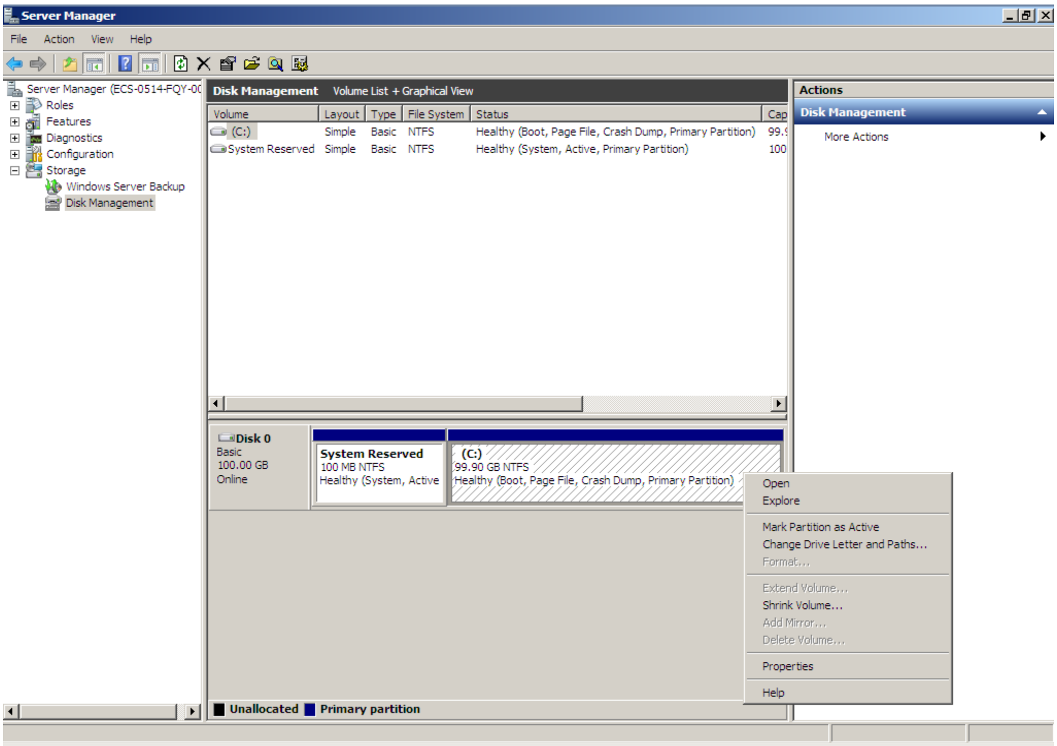
The **Disk Management** window is displayed.

Figure 5-16 Refresh (shrink volume)



Step 3 In the **(C:)** area of **Disk 0**, right-click the blank area and choose **Shrink Volume**.

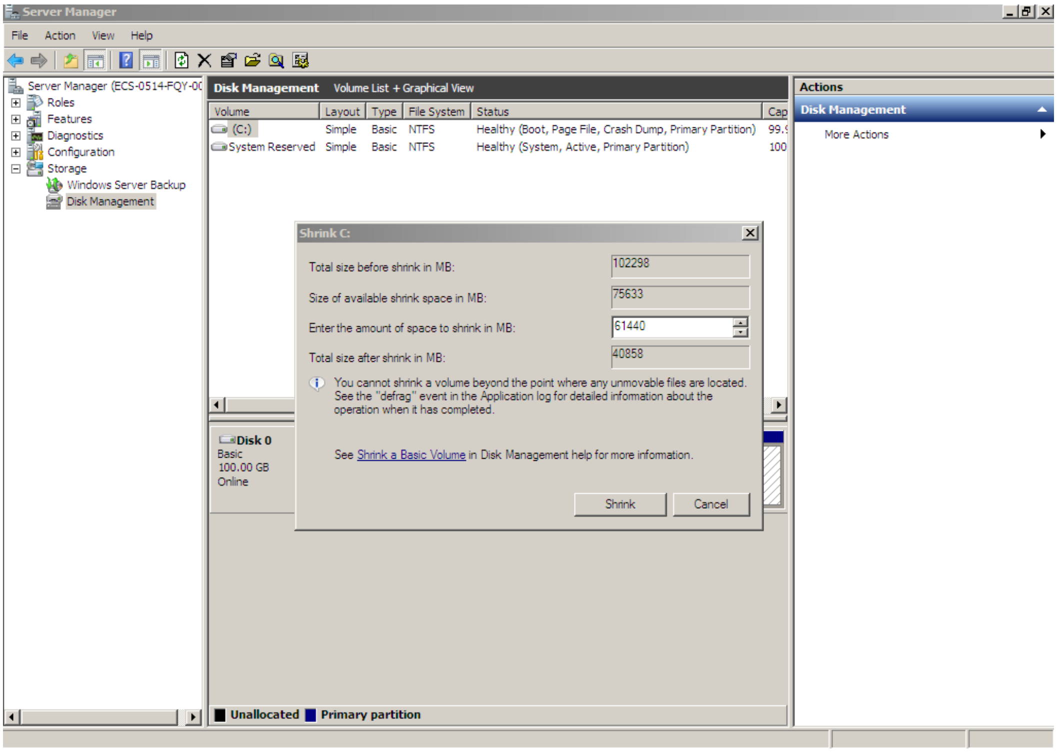
Figure 5-17 Shrink Volume



Step 4 The system automatically queries the available shrink space. In the displayed dialog box, enter the available space and click **Shrink**.

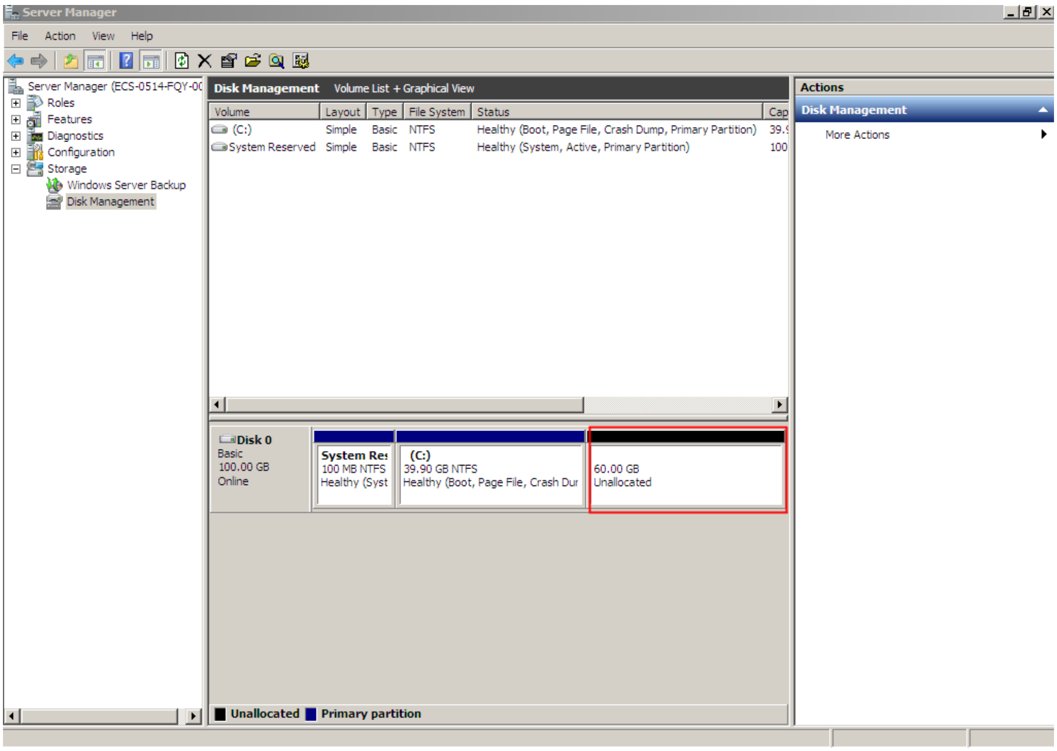
In this example, the volume available space is 60 GiB. Therefore, enter **61440** (60 × 1024 MiB).

Figure 5-18 Shrink (shrink volume)



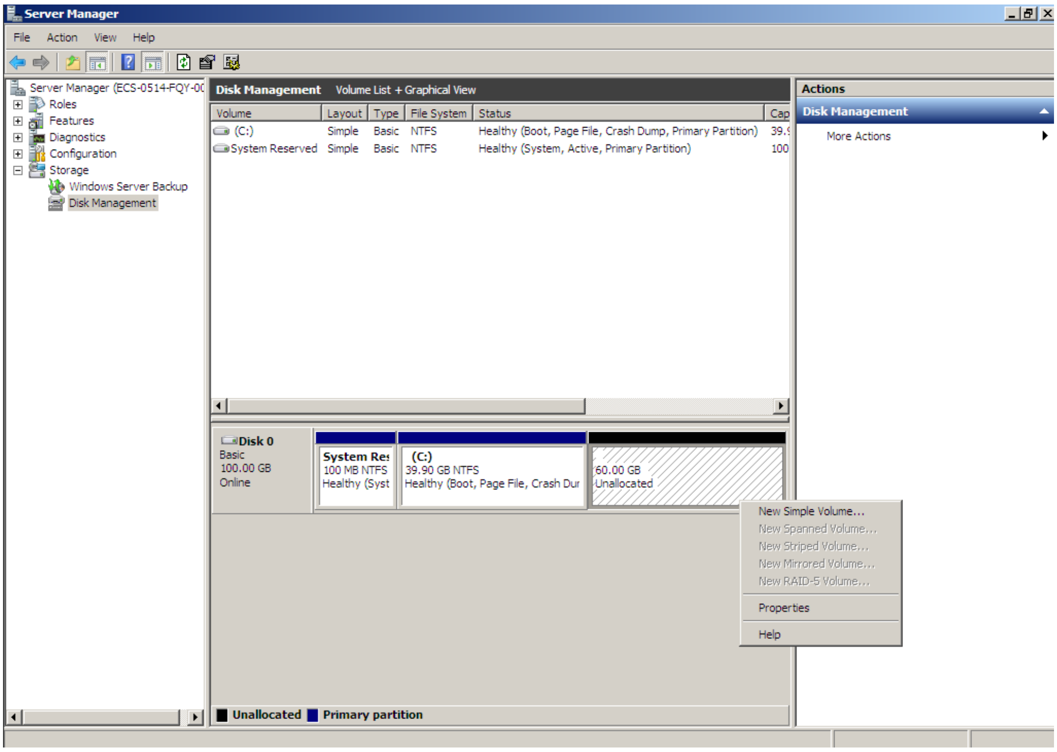
After the operation is complete, **Disk 0** has 60 GiB unallocated space.

Figure 5-19 Unallocated (shrink volume)



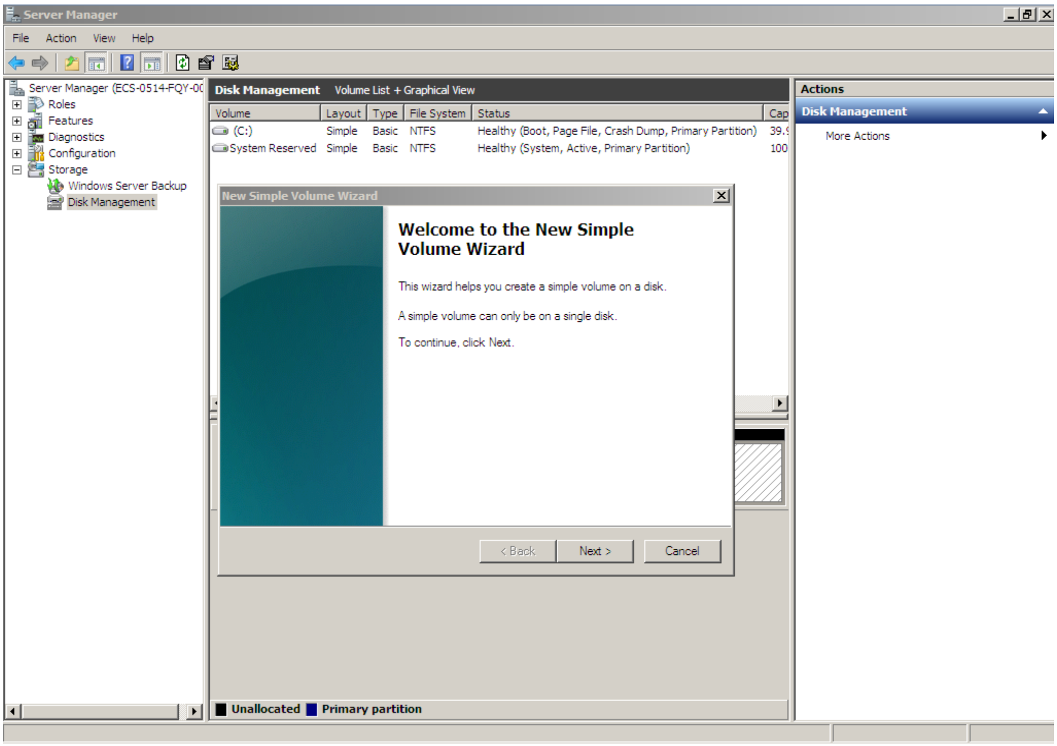
Step 5 In the **Unallocated** area of **Disk 0**, right-click the blank area and choose **New Simple Volume**.

Figure 5-20 New Simple Volume (shrink volume)



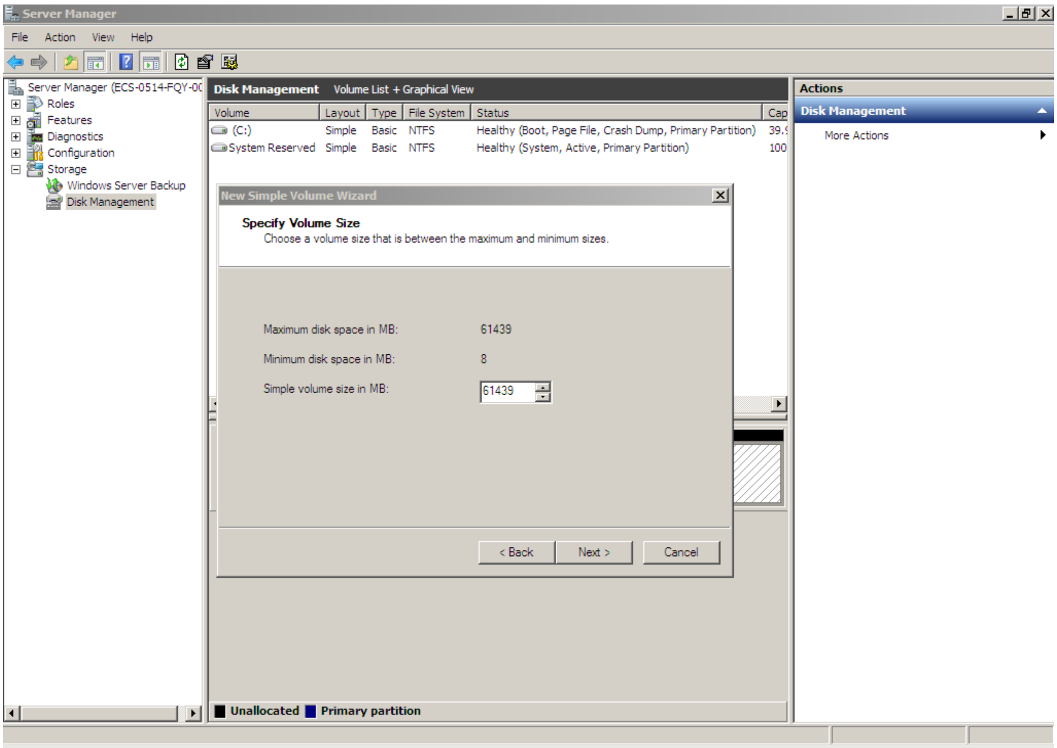
Step 6 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 5-21 New Simple Volume Wizard (shrink volume)



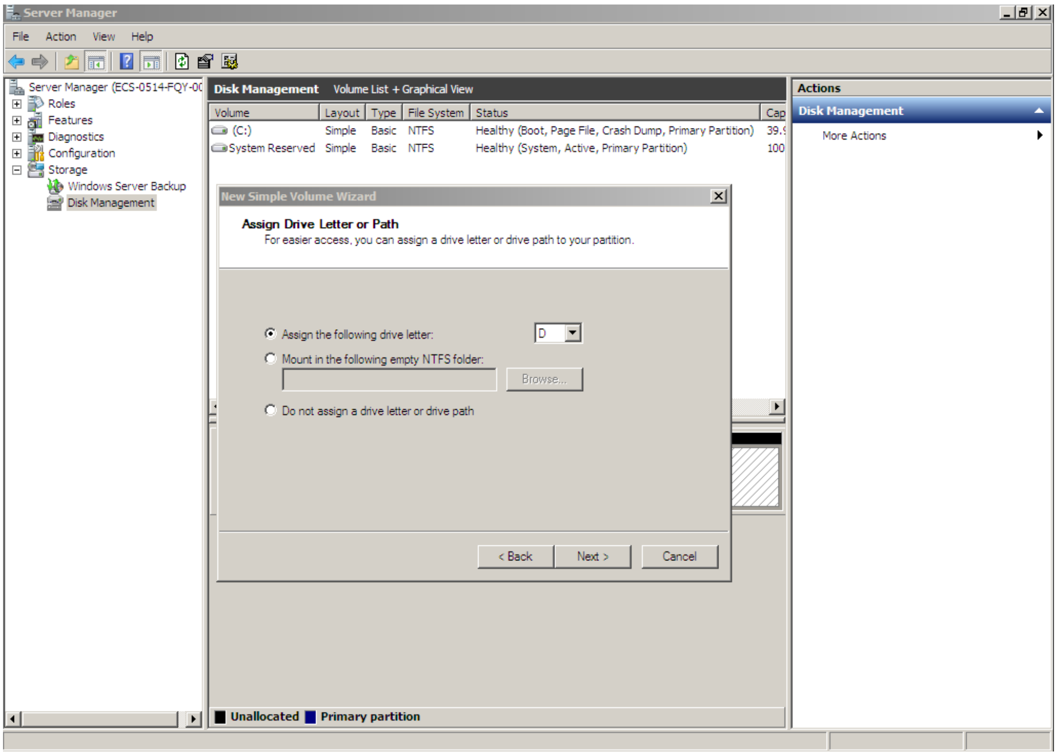
Step 7 On the displayed **Specify Volume Size** page, set **Simple volume size in MB** and click **Next**. In this example, the default size is used.

Figure 5-22 Specify Volume Size (shrink volume)



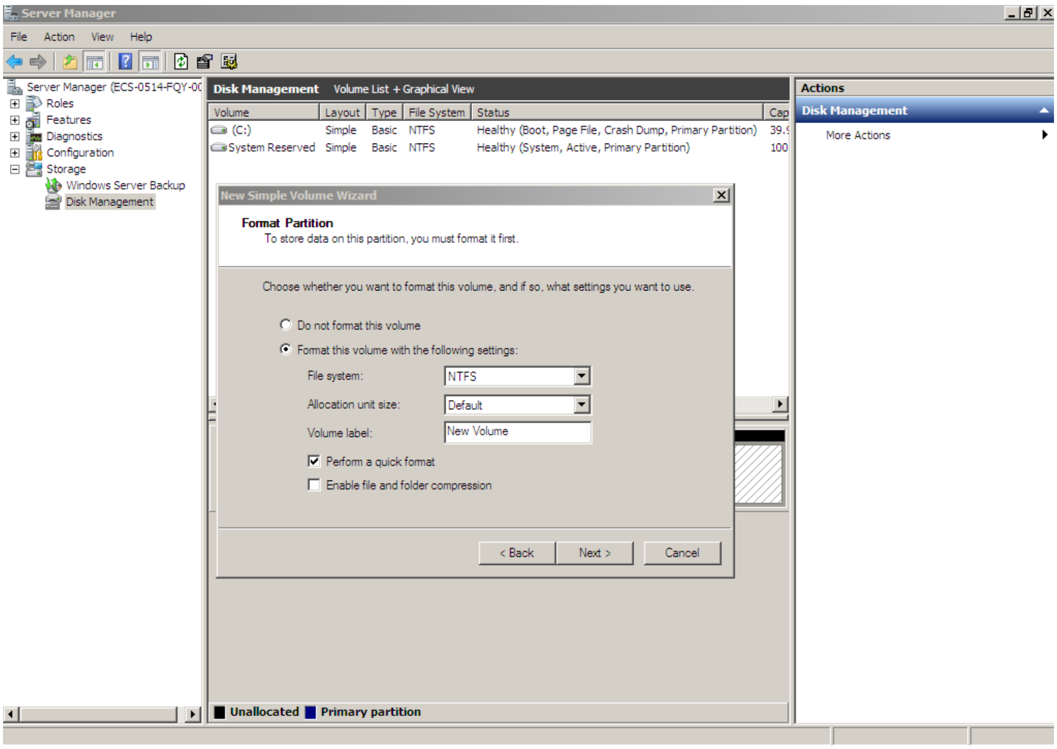
Step 8 On the displayed **Assign Drive Letter and Path** page, click **Assign the following drive letter**, select a drive letter, and click **Next**. In this example, drive letter **D** is selected.

Figure 5-23 Assign Drive Letter or Path (shrink volume)



Step 9 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

Figure 5-24 Format Partition (shrink volume)



Step 10 Click **Finish**.

After the expansion succeeded, new volume (D:) is displayed.

Figure 5-25 Completing the New Simple Volume Wizard (new volume D:)

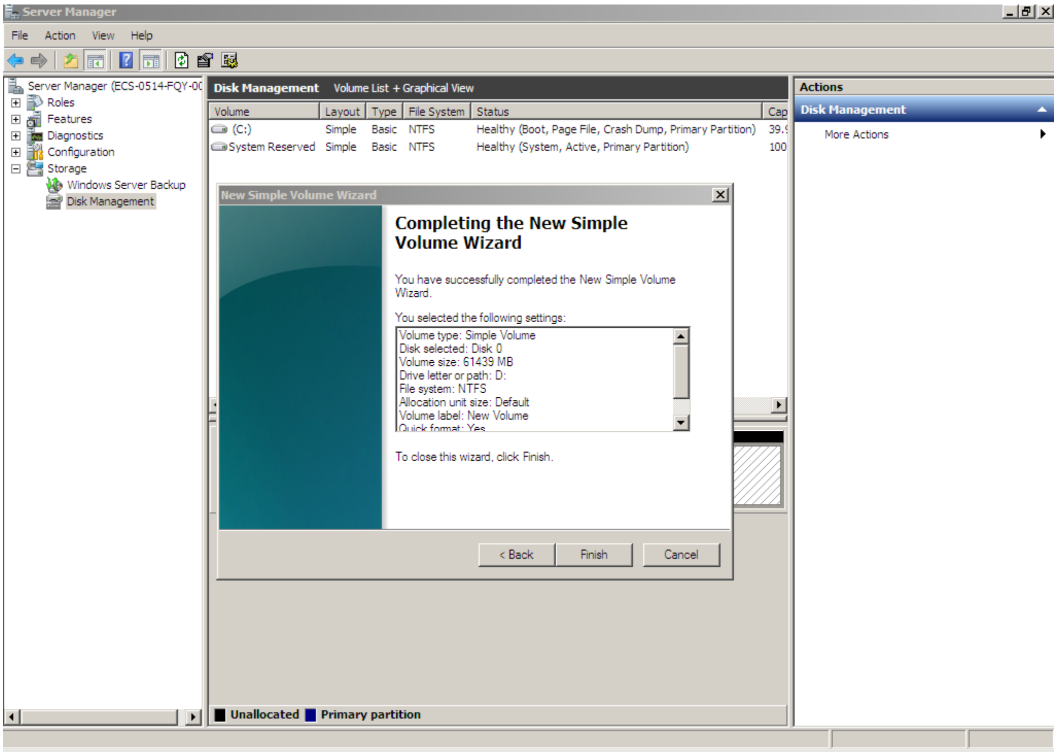
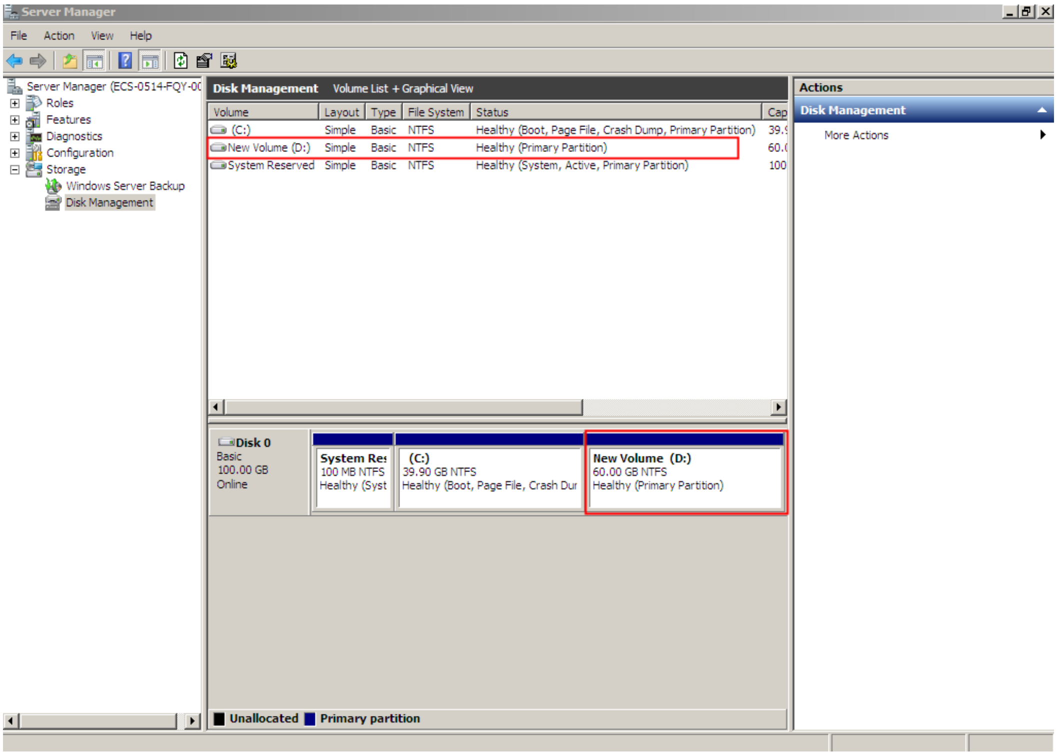


Figure 5-26 New Volume (D:)



----End

Data Disk: Add the Additional Space to the Original Volume

In this example, the data disk has 100 GiB originally, and 50 GiB is added on the management console. The following procedure describes how to add this 50 GiB to volume (D:) on the server. After the operation is complete, volume (D:) has 150 GiB of capacity and can be used as a data volume.

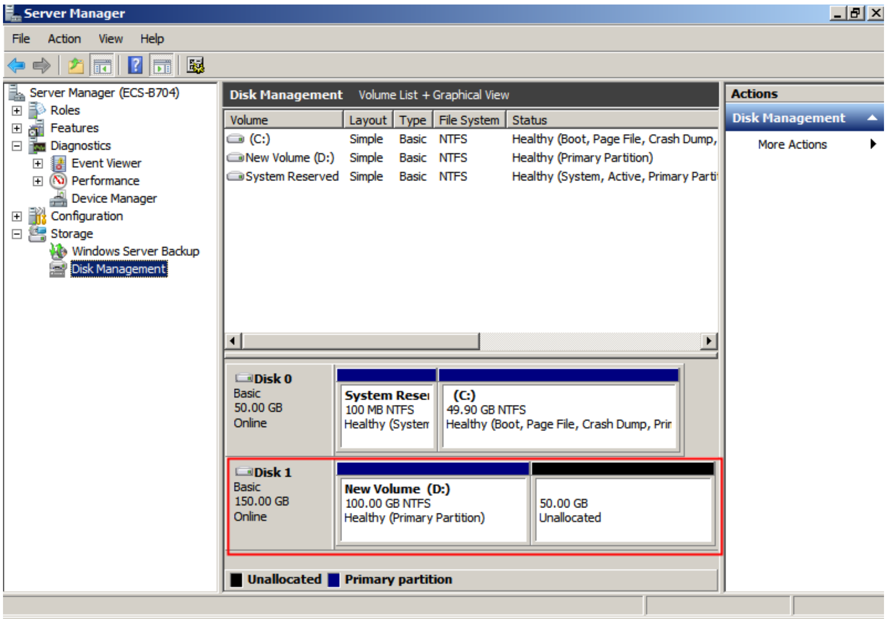
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-27 Disk Management (data disk)



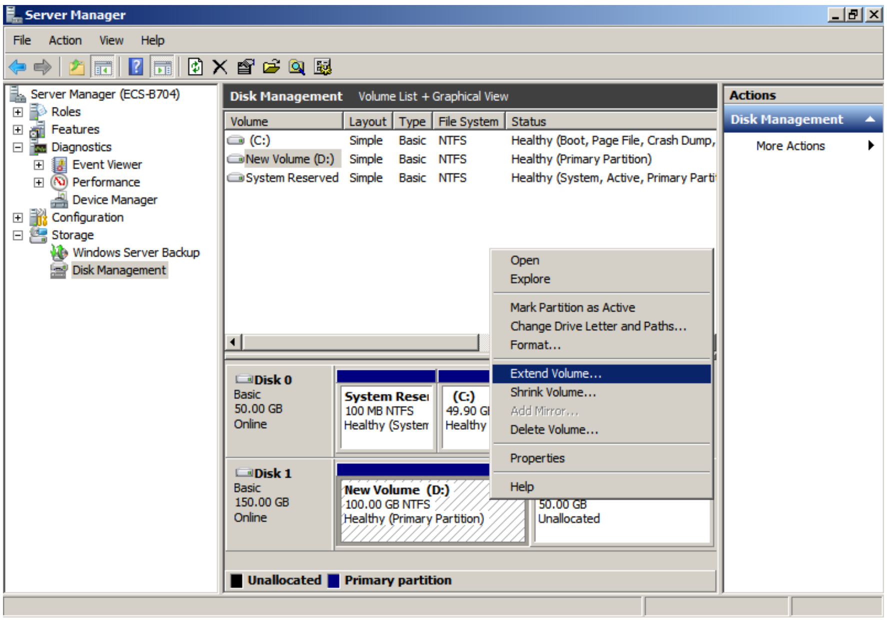
NOTE

If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

Step 3 On the **Disk Management** page, select the disk and volume that you want to extend. The current volume size and unallocated space are displayed.

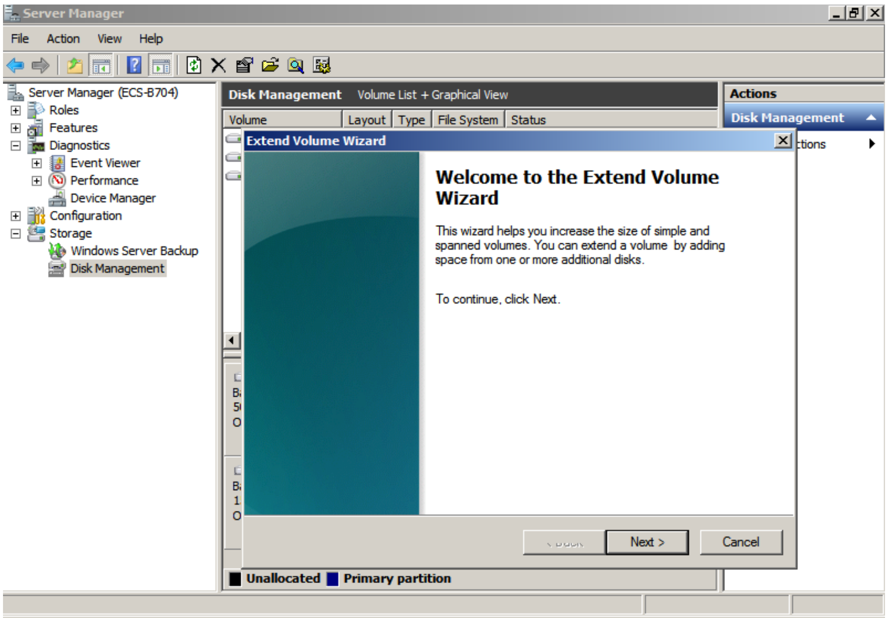
Step 4 Right-click the target volume and choose **Extend Volume**.

Figure 5-28 Choosing Extend Volume (Windows Server 2008)



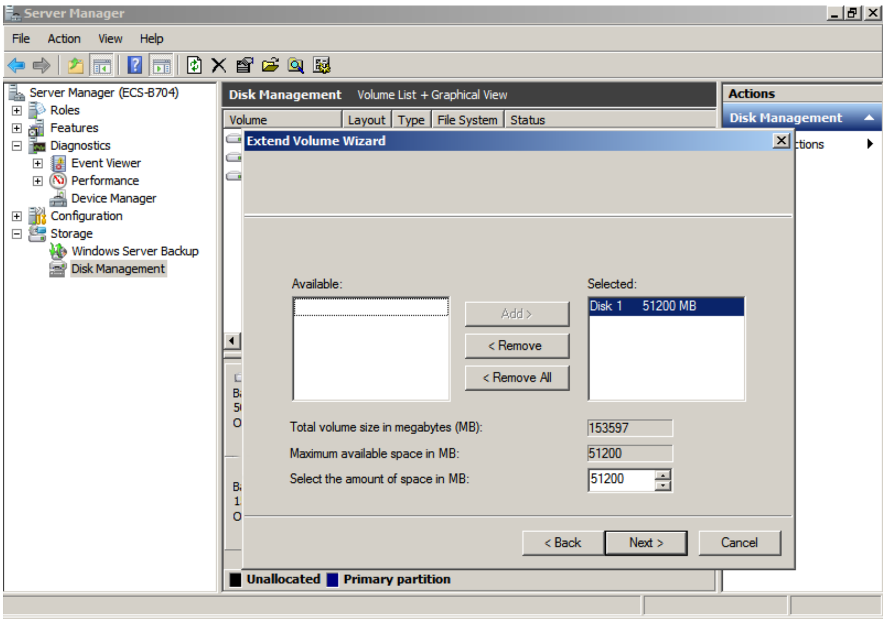
Step 5 On the displayed **Extend Volume Wizard** window, click **Next**.

Figure 5-29 Extend Volume Wizard (Windows Server 2008)



Step 6 In the text box to the right of **Select the amount of space in MB**, enter the amount of the additional space and click **Next**.

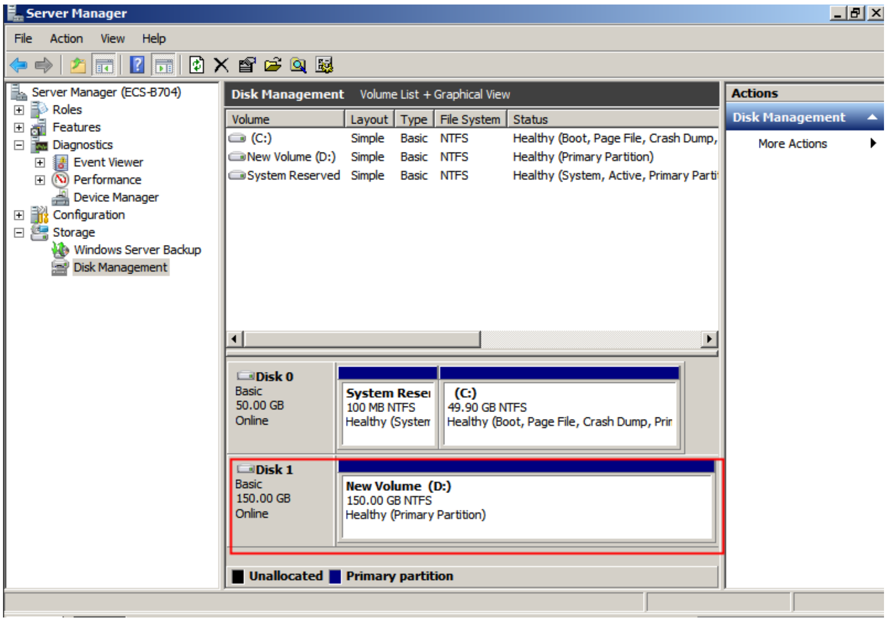
Figure 5-30 Selecting space (Windows Server 2008)



Step 7 Click **Finish**.

After the expansion succeeded, the partition size is larger than the original size.

Figure 5-31 Capacity expansion succeeded (Windows Server 2008)



----End

Data Disk: Create a New Volume with the Additional Space

In this example, the data disk has 40 GiB originally, and 60 GiB is added on the management console. The following procedure describes how to use this 60 GiB to create a new volume, for example volume (E:), on the server. After the operation is complete, new volume (E:) has 60 GiB of capacity and can be used as a data volume.

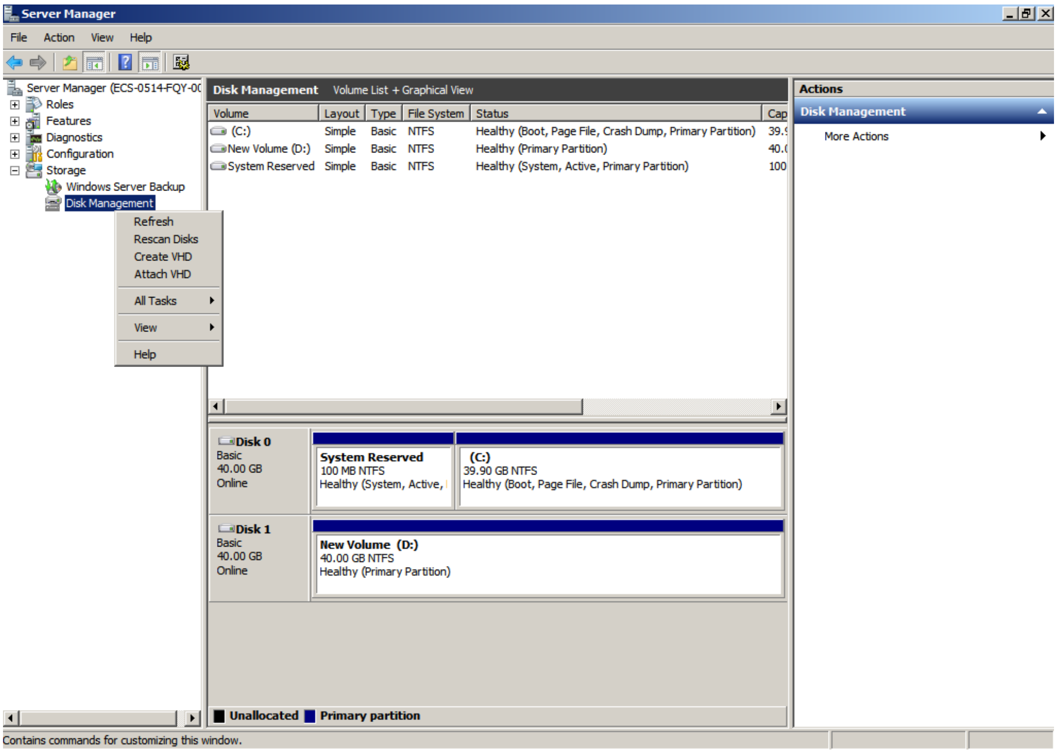
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The **Disk Management** window is displayed.

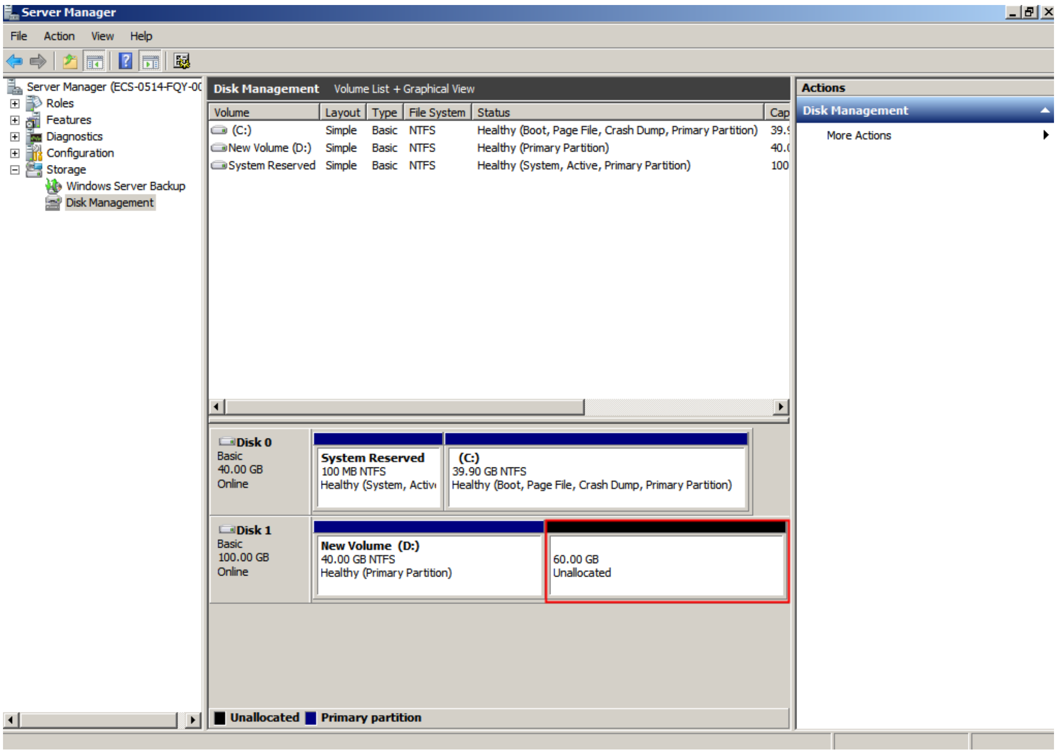
Figure 5-32 Refresh (data disk)



Step 3 If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

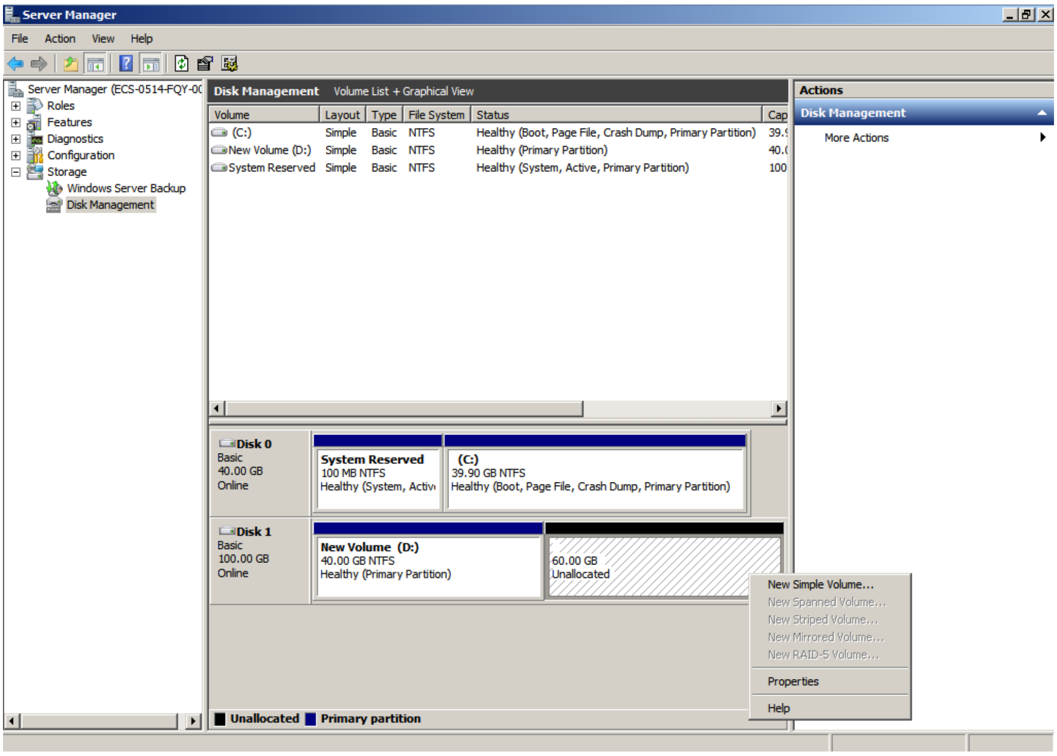
After the refresh, the additional space is displayed in the right area and is unallocated.

Figure 5-33 Unallocated disk space (data disk)



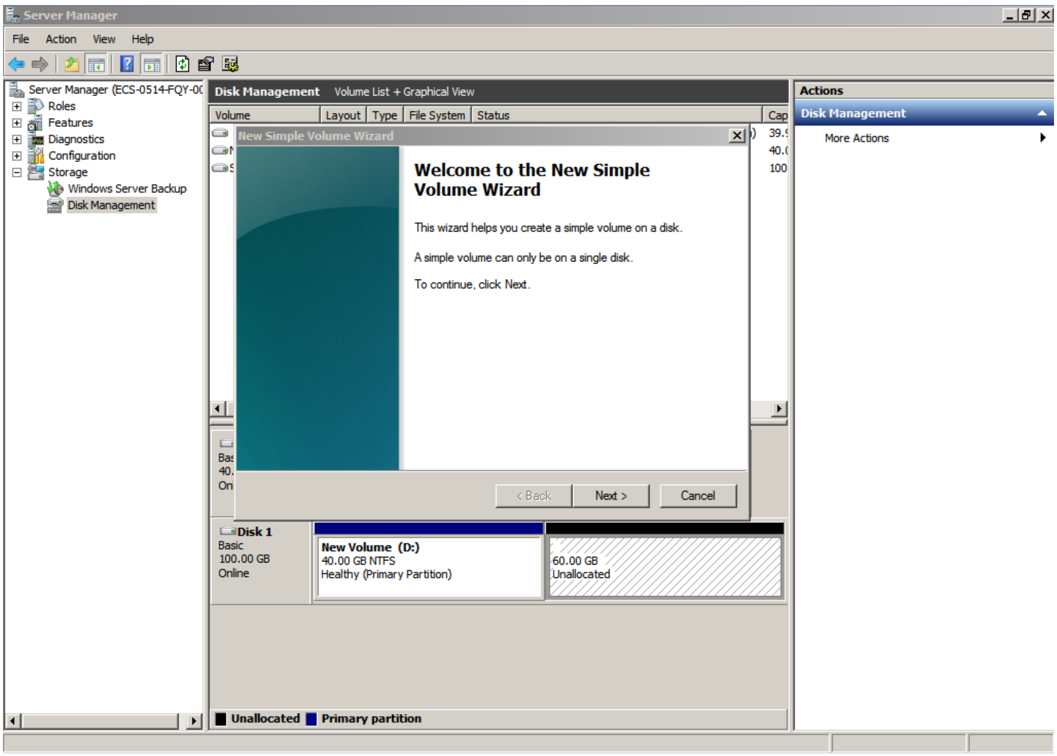
Step 4 In the **Unallocated** area of **Disk 1**, right-click the blank area and choose **New Simple Volume**.

Figure 5-34 New Simple Volume (data disk)



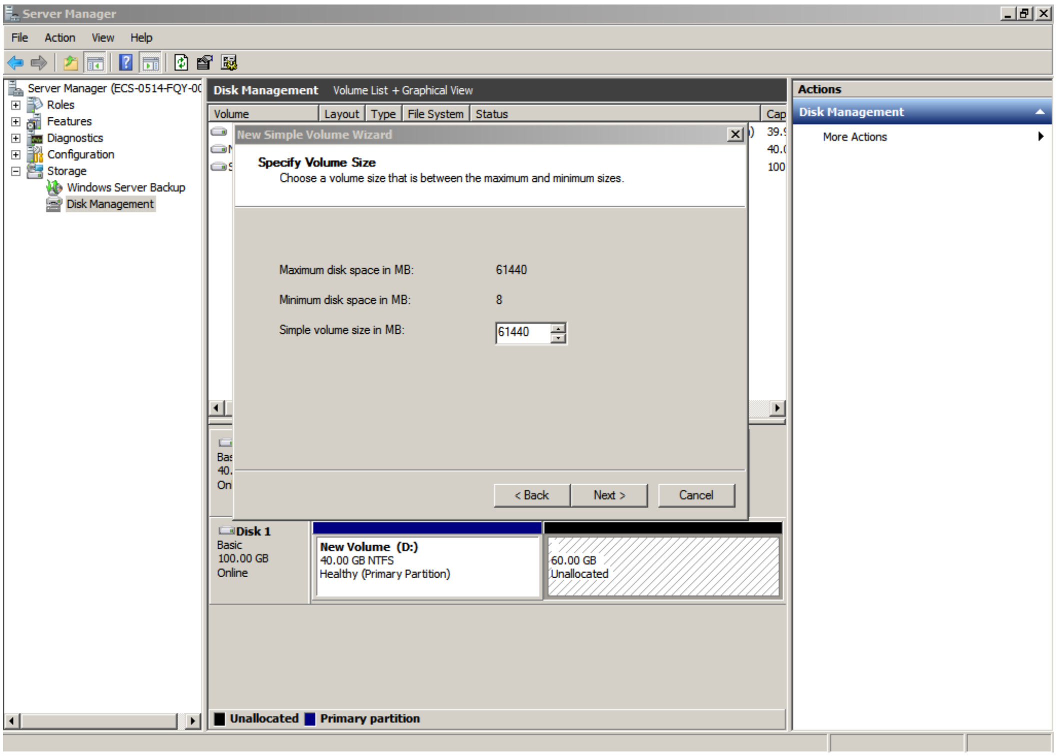
Step 5 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 5-35 New Simple Volume Wizard (data disk)



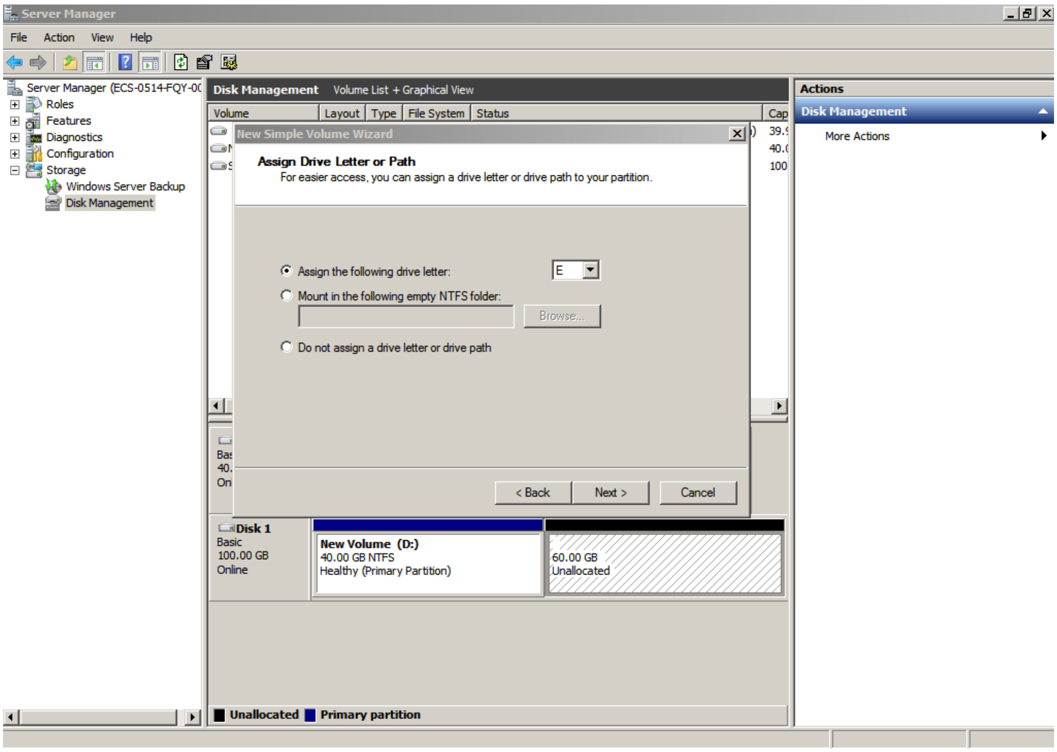
Step 6 On the displayed **Specify Volume Size** page, set **Simple volume size in MB** and click **Next**. In this example, the default size is used.

Figure 5-36 Specify Volume Size (data disk)



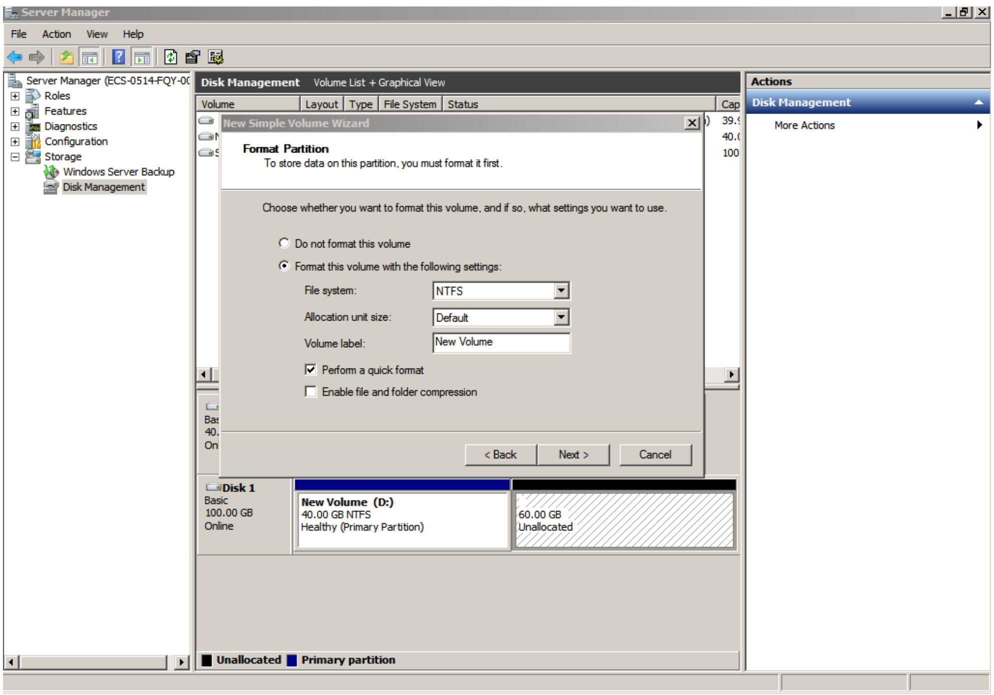
Step 7 On the displayed **Assign Drive Letter and Path** page, click **Assign the following drive letter**, select a drive letter, and click **Next**. In this example, drive letter **E** is selected.

Figure 5-37 Assign Drive Letter or Path (data disk)



Step 8 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

Figure 5-38 Format Partition (data disk)



Step 9 Click **Finish**.

After the expansion succeeded, new volume (E:) is displayed.

Figure 5-39 Completing the New Simple Volume Wizard (new volume E:)

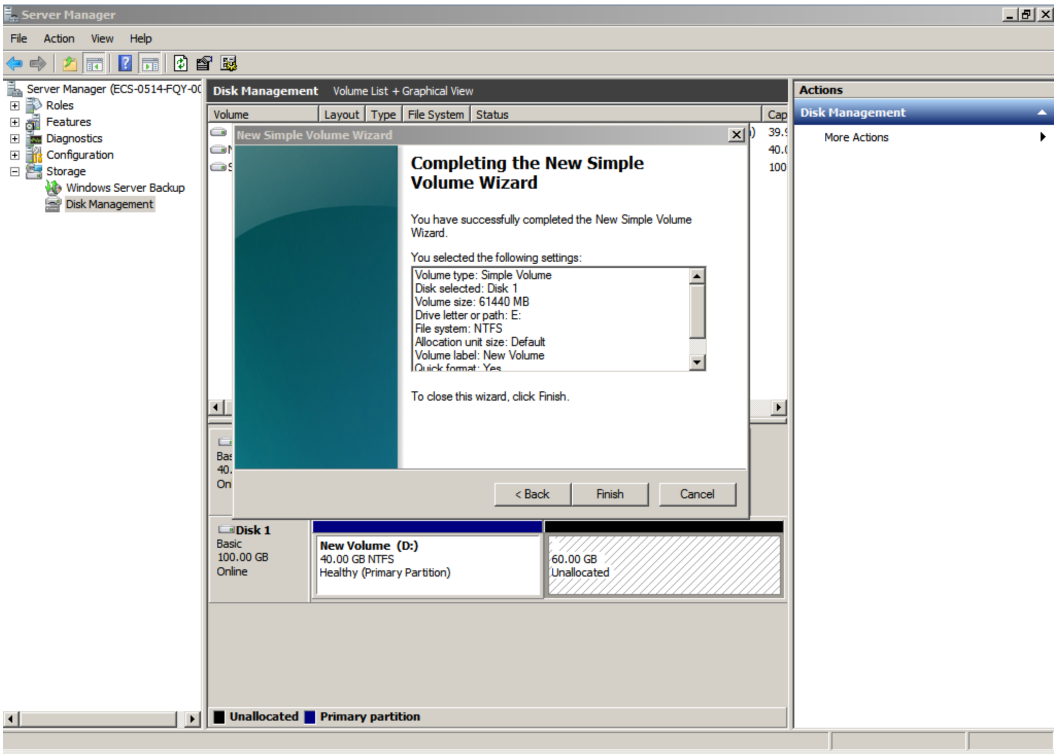
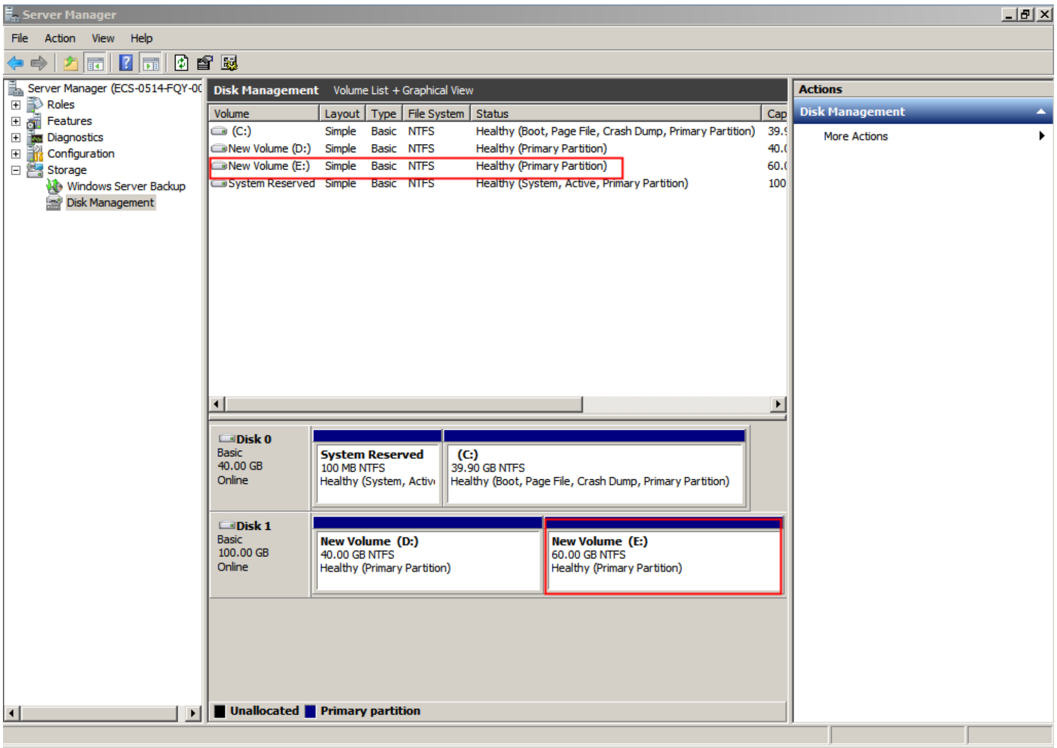


Figure 5-40 New Volume (E:)



----End

5.5 Extending Disk Partitions and File Systems (Windows Server 2016)

Scenarios

After a disk is expanded on the management console, the disk size is enlarged, but the additional space cannot be used directly.

In Windows, you must allocate the additional space to an existing partition or a new partition.

If the disk capacity is expanded on a stopped server, the additional space of a Windows system disk or Windows data disk will be automatically added to the partition at the end of the disk upon the server startup. In this case, the additional space can be used directly.

This section uses Windows Server 2016 Standard 64bit as the sample OS to describe the expansion methods:

- For a system disk:
  - If volume (C:) already exists, you can add the additional space to volume (C:) and use it as a system volume. For details, see **System Disk: Add the Additional Space to the Original Volume**.
  - If volume (C:) already exists, you can create a new volume such as volume (F:) with the additional space and use the new volume as a data volume. For details, see **System Disk: Create a New Volume with the Additional Space**.
- For a data disk:
  - If volume (D:) already exists, you can add the additional space to volume (D:) and use it as a data volume. For details, see **Data Disk: Add the Additional Space to the Original Volume**.
  - If volume (D:) already exists, you can create a new volume such as volume (E:) with the additional space and use the new volume as a data volume. For details, see **Data Disk: Create a New Volume with the Additional Space**.

The method for allocating the additional space varies with the server OS. This section is used for reference only. For detailed operations and differences, see the corresponding OS documents.

NOTICE

Performing the expansion operations with caution. Incorrect operations may lead to data loss or exceptions, so you are advised to back up the disk data using CBR or snapshots before expansion. For details about using CBR, see **Managing EVS Backups**. For details about using snapshots, see the *Elastic Volume Service API Reference*

Prerequisites

- You have expanded the disk capacity and attached the disk to a server on the management console. For details, see **Expanding Capacity for an In-use EVS Disk** or **Expanding Capacity for an Available EVS Disk**.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

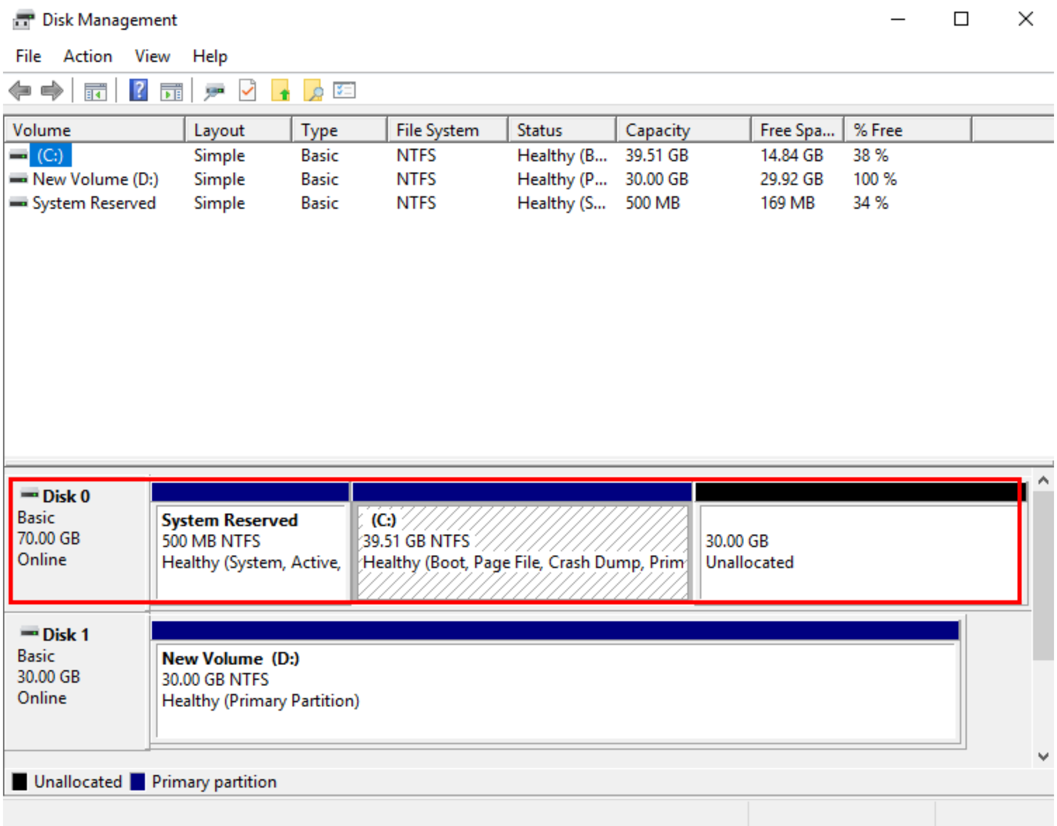
System Disk: Add the Additional Space to the Original Volume

In this example, the system disk has 40 GiB originally, and 30 GiB is added on the management console. The following procedure describes how to add this 30 GiB to volume (C:) on the server. After the operation is complete, volume (C:) will have 70 GiB of capacity and can be used as a system volume.

Step 1 On the desktop of the server, right-click the start icon in lower left corner and choose **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-41 Disk Management (Windows Server 2016)



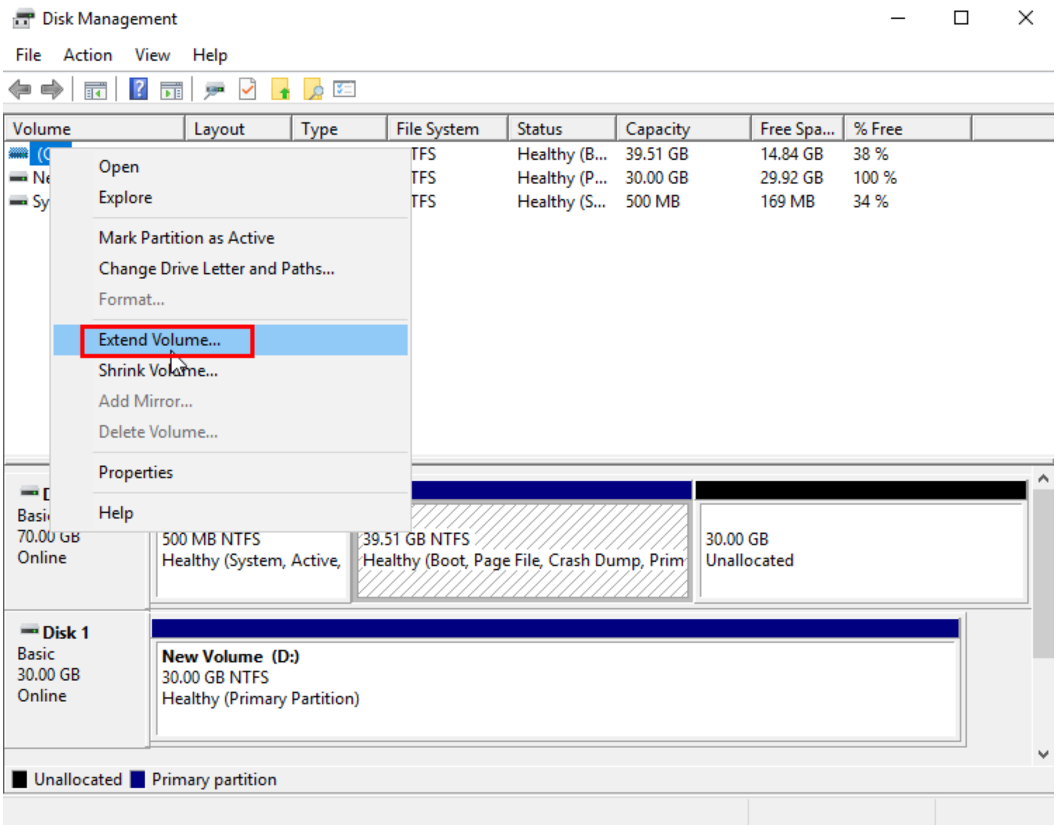
NOTE

If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

Step 2 On the **Disk Management** page, select the disk and volume that you want to extend. The current volume size and unallocated space are displayed.

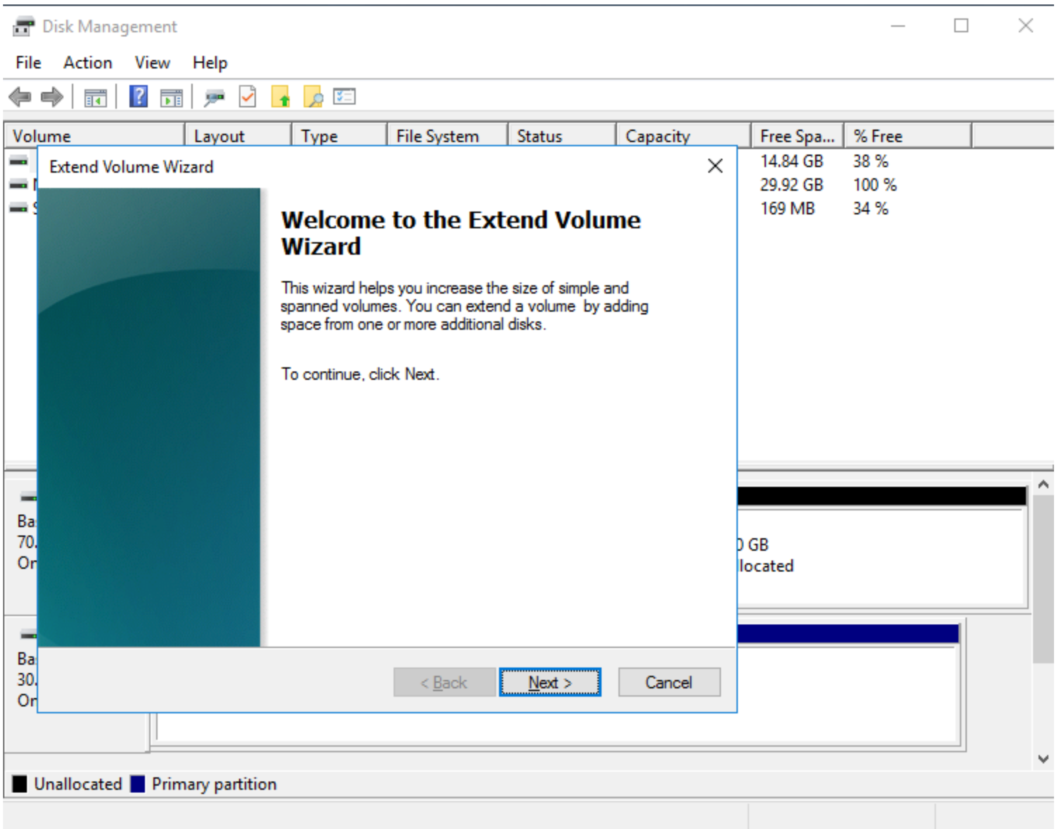
Step 3 Right-click the target volume and choose **Extend Volume**.

Figure 5-42 Choosing Extend Volume (Windows Server 2016)



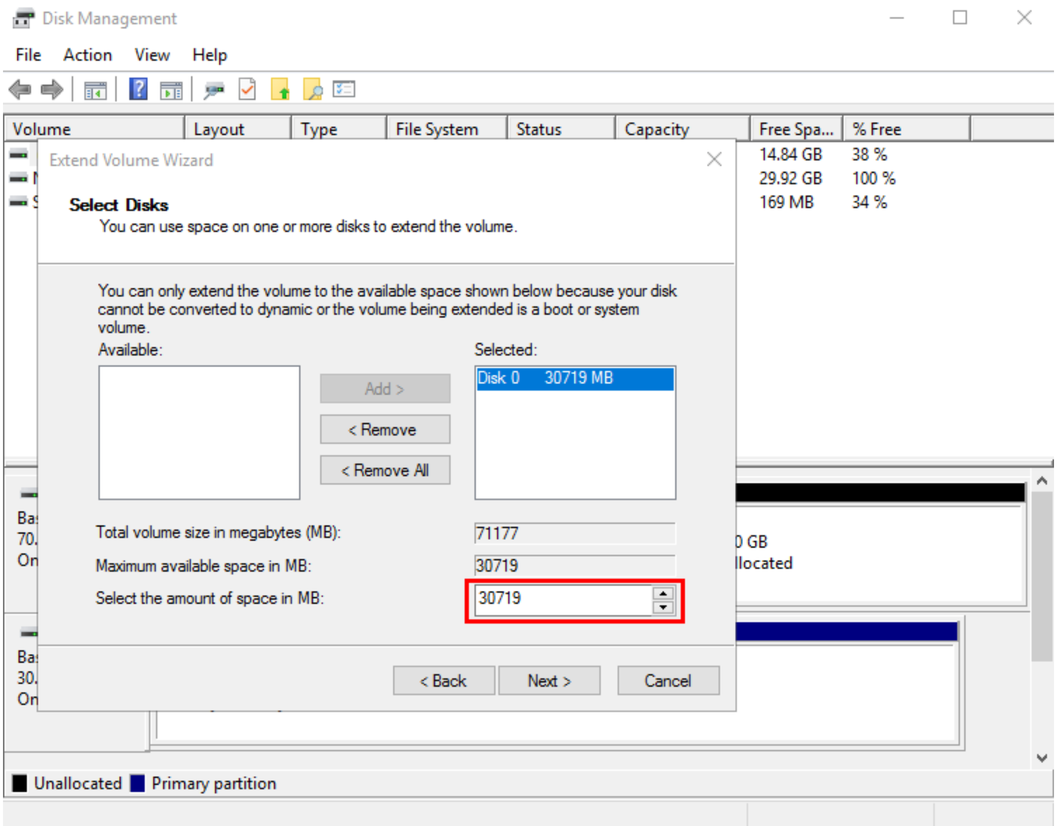
Step 4 On the displayed **Extend Volume Wizard** window, click **Next**.

Figure 5-43 Extend Volume Wizard (Windows Server 2016)



Step 5 In the text box to the right of **Select the amount of space in MB**, enter the amount of the additional space and click **Next**.

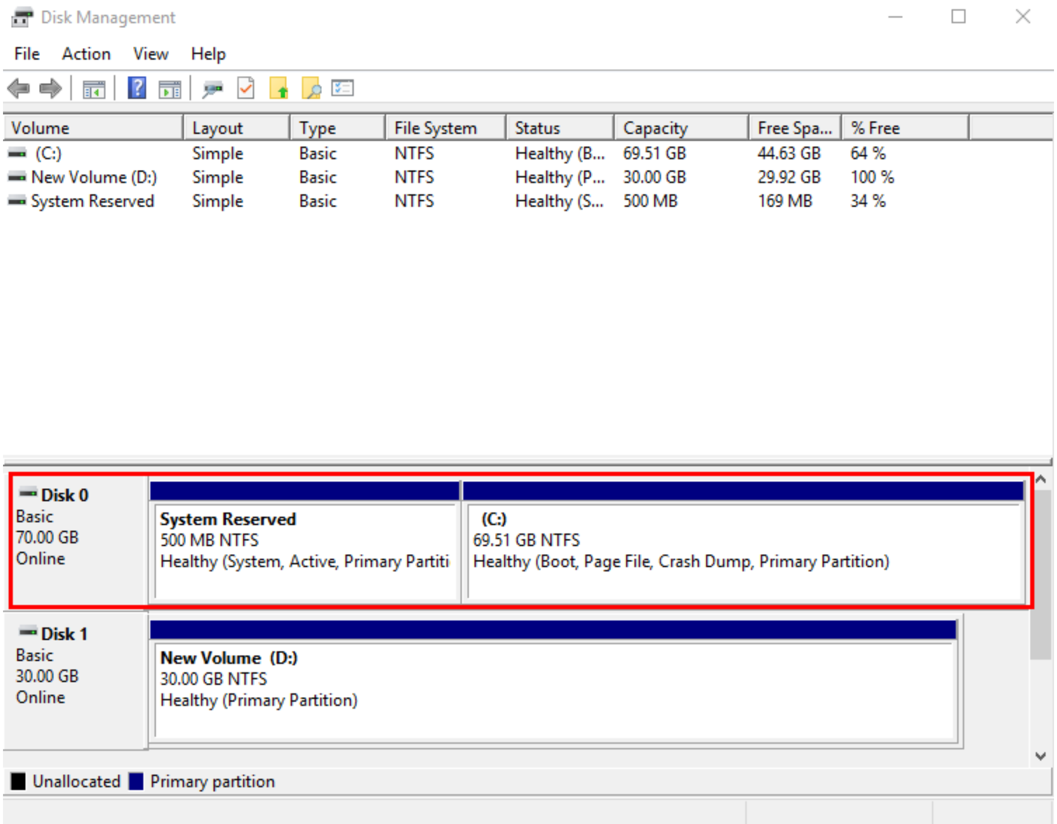
Figure 5-44 Selecting space (Windows Server 2016)



Step 6 Click **Finish**.

After the expansion succeeded, the partition size is larger than the original size.

Figure 5-45 Capacity expansion succeeded (Windows Server 2016)



----End

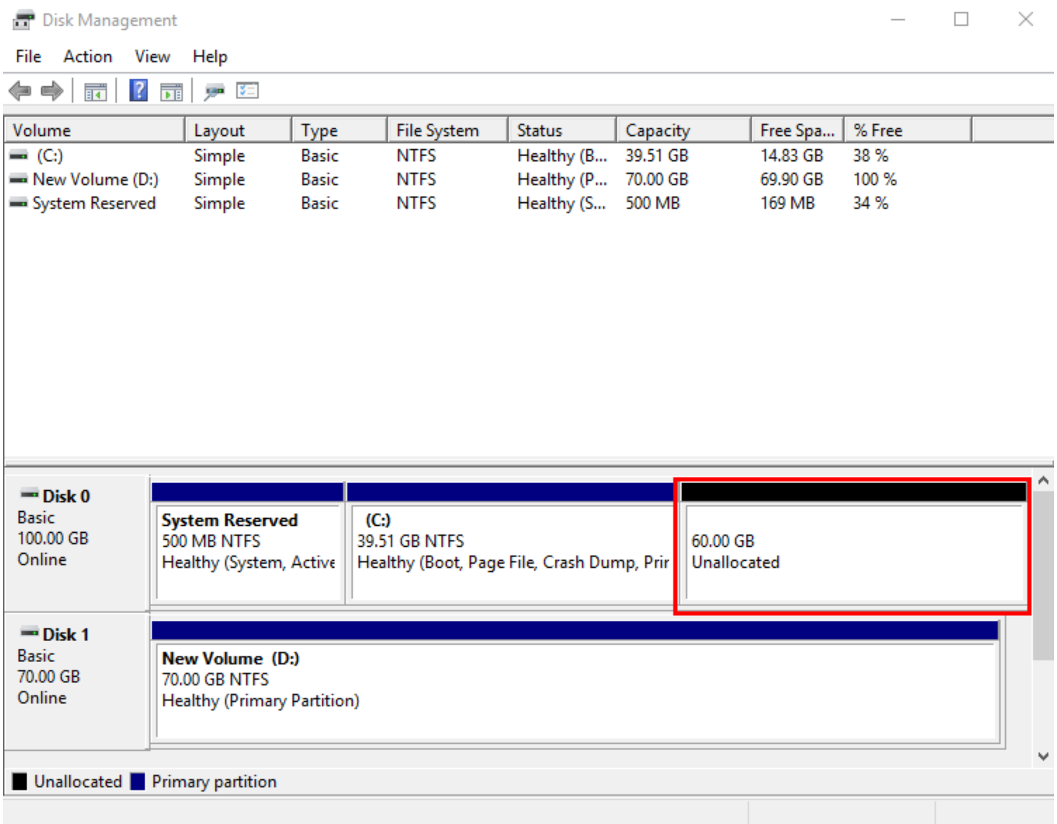
System Disk: Create a New Volume with the Additional Space

In this example, the system disk has 40 GiB originally, and 60 GiB is added on the management console. The following procedure describes how to use this 60 GiB to create a new volume, for example volume (F:), on the server. After the operation is complete, new volume (F:) has 60 GiB of capacity and can be used as a data volume.

Step 1 On the desktop of the server, right-click the start icon in lower left corner and choose **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-46 Unallocated disk space (Windows Server 2016 system disk)

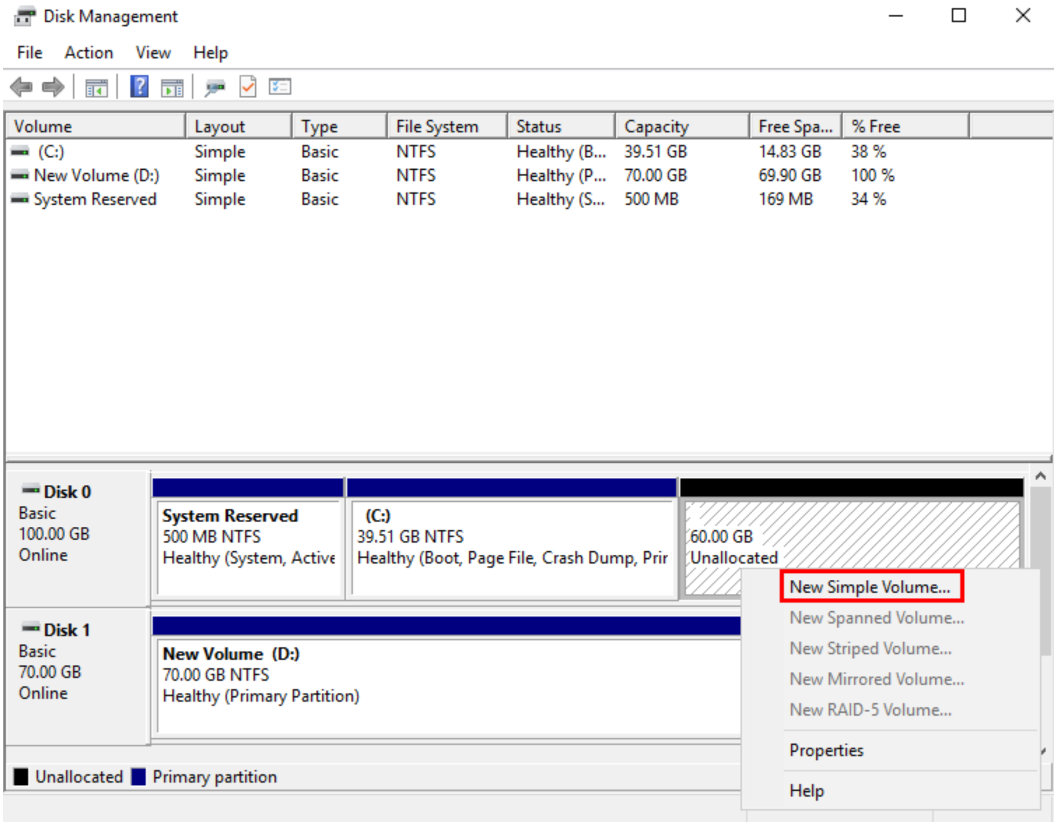


NOTE

If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

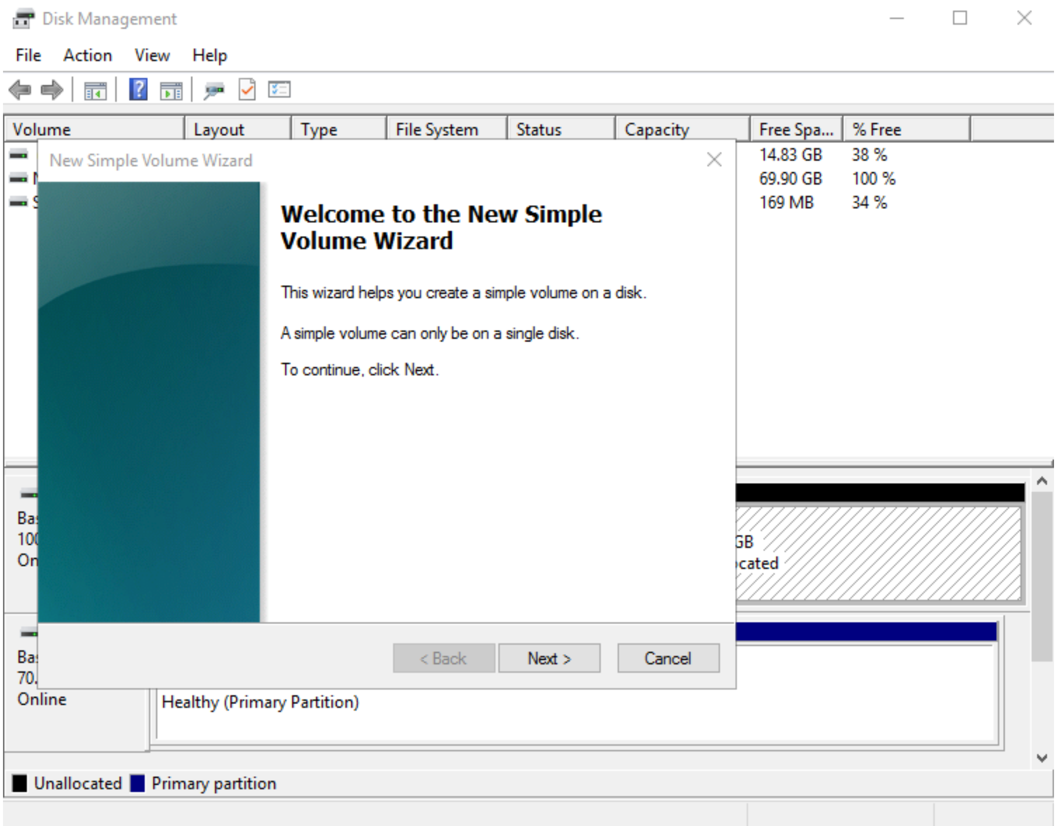
Step 2 In the **Unallocated** area of **Disk 0**, right-click the blank area and choose **New Simple Volume**.

Figure 5-47 New Simple Volume (Windows Server 2016 system disk)



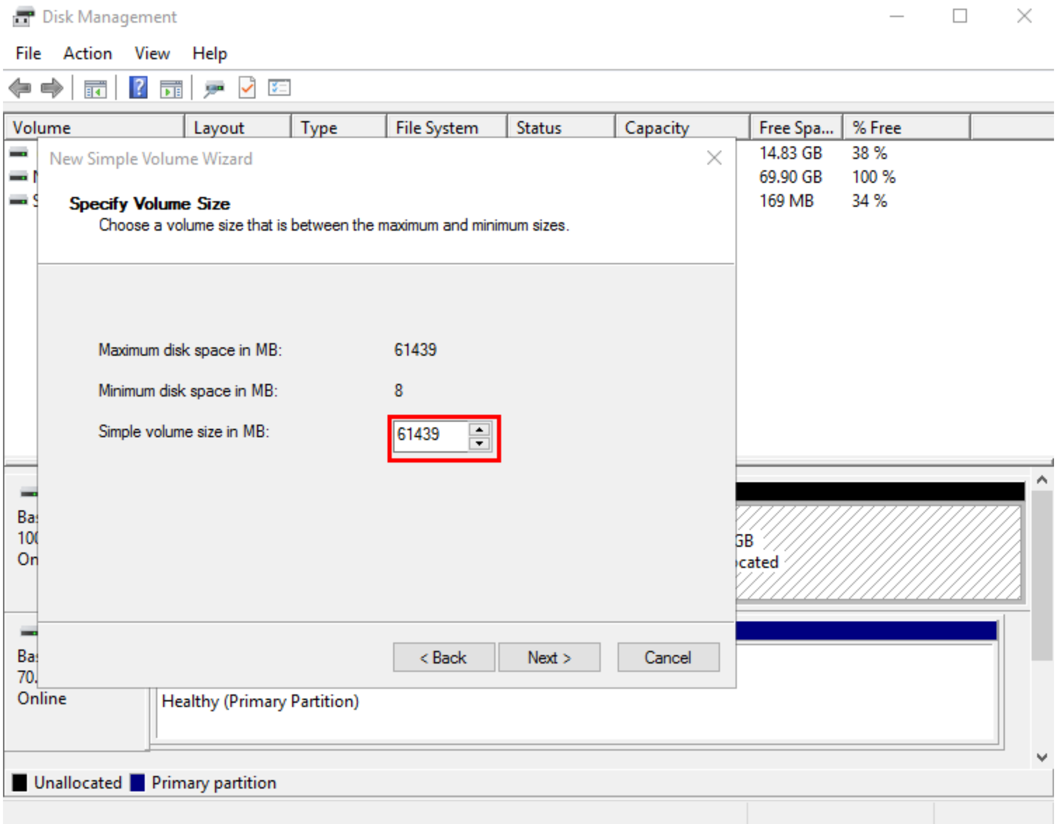
Step 3 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 5-48 New Simple Volume Wizard (Windows Server 2016 system disk)



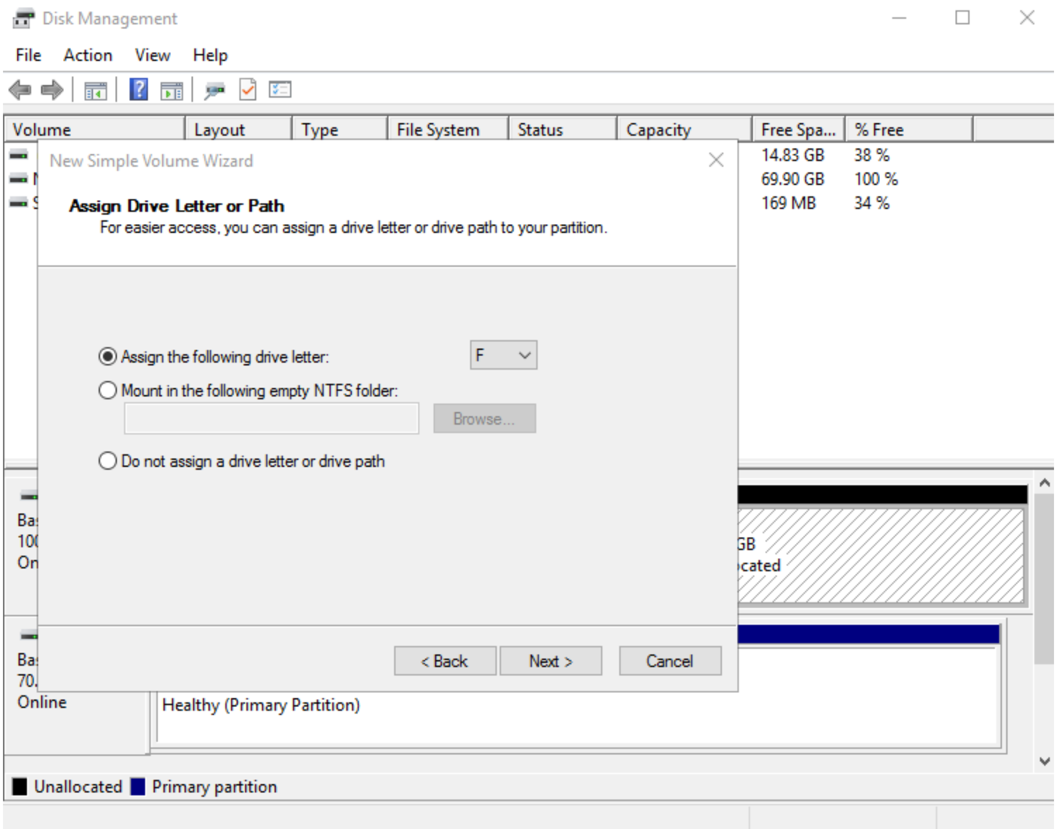
Step 4 On the displayed **Specify Volume Size** page, set **Simple volume size in MB** and click **Next**. In this example, the default size is used.

Figure 5-49 Specify Volume Size (Windows Server 2016 system disk)



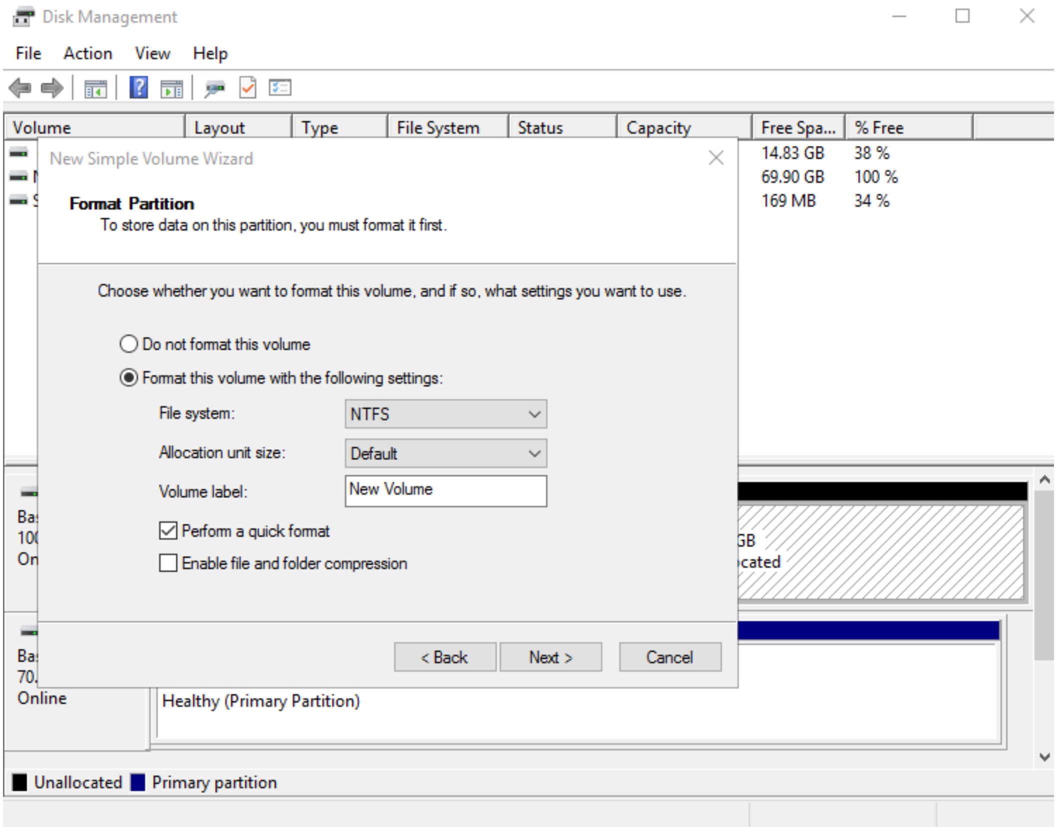
Step 5 On the displayed **Assign Drive Letter and Path** page, click **Assign the following drive letter**, select a drive letter, and click **Next**. In this example, drive letter **F** is selected.

Figure 5-50 Assign Drive Letter or Path (Windows Server 2016 system disk)



Step 6 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

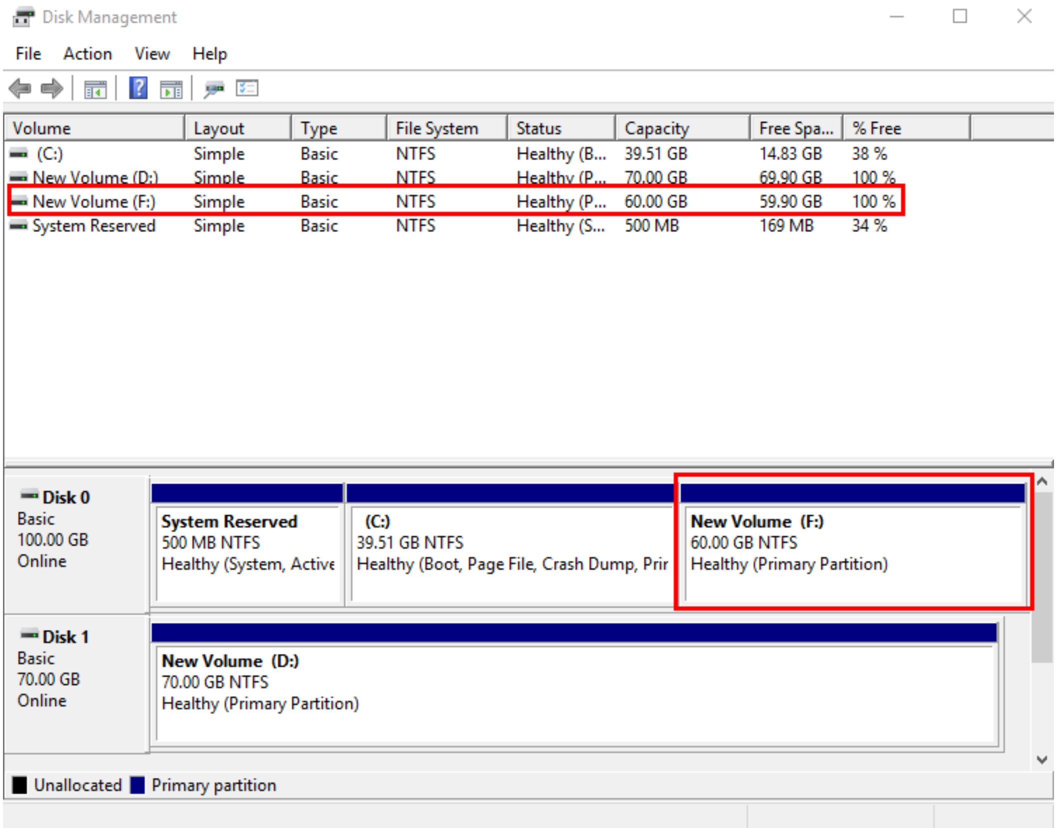
Figure 5-51 Format Partition (Windows Server 2016 system disk)



Step 7 Click **Finish**.

After the expansion succeeded, new volume (F:) is displayed.

Figure 5-52 Volume (F:) (Windows Server 2016)



----End

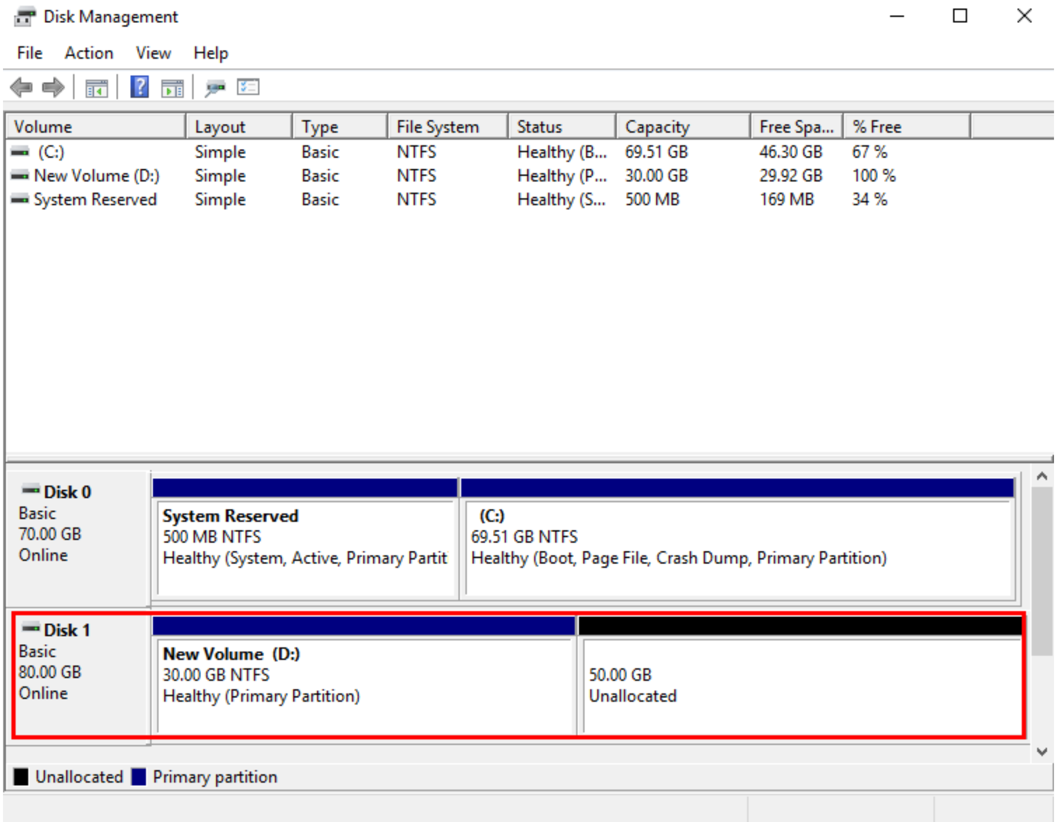
Data Disk: Add the Additional Space to the Original Volume

In this example, the data disk has 30 GiB originally, and 50 GiB is added on the management console. The following procedure describes how to add this 50 GiB to volume (D:) on the server. After the operation is complete, volume (D:) has 80 GiB of capacity and can be used as a data volume.

Step 1 On the desktop of the server, right-click the start icon in lower left corner and choose **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-53 Disk Management (Windows Server 2016 data disk)



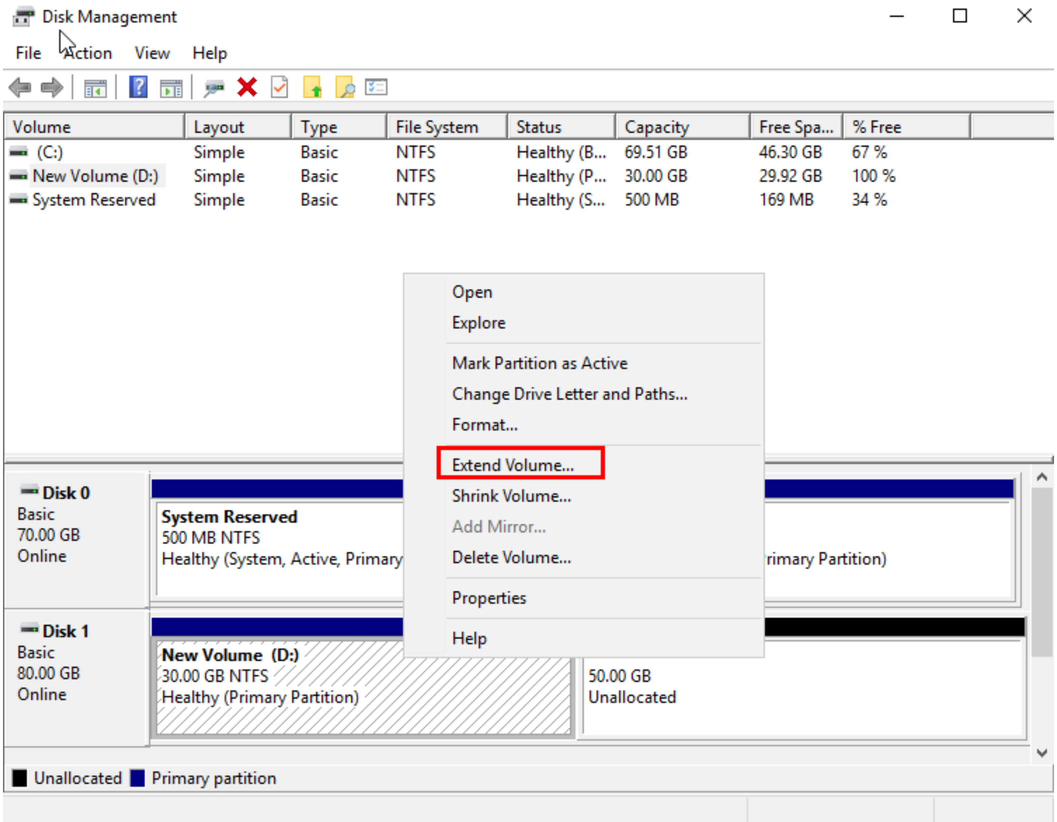
NOTE

If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

Step 2 On the **Disk Management** page, select the disk and volume that you want to extend. The current volume size and unallocated space are displayed.

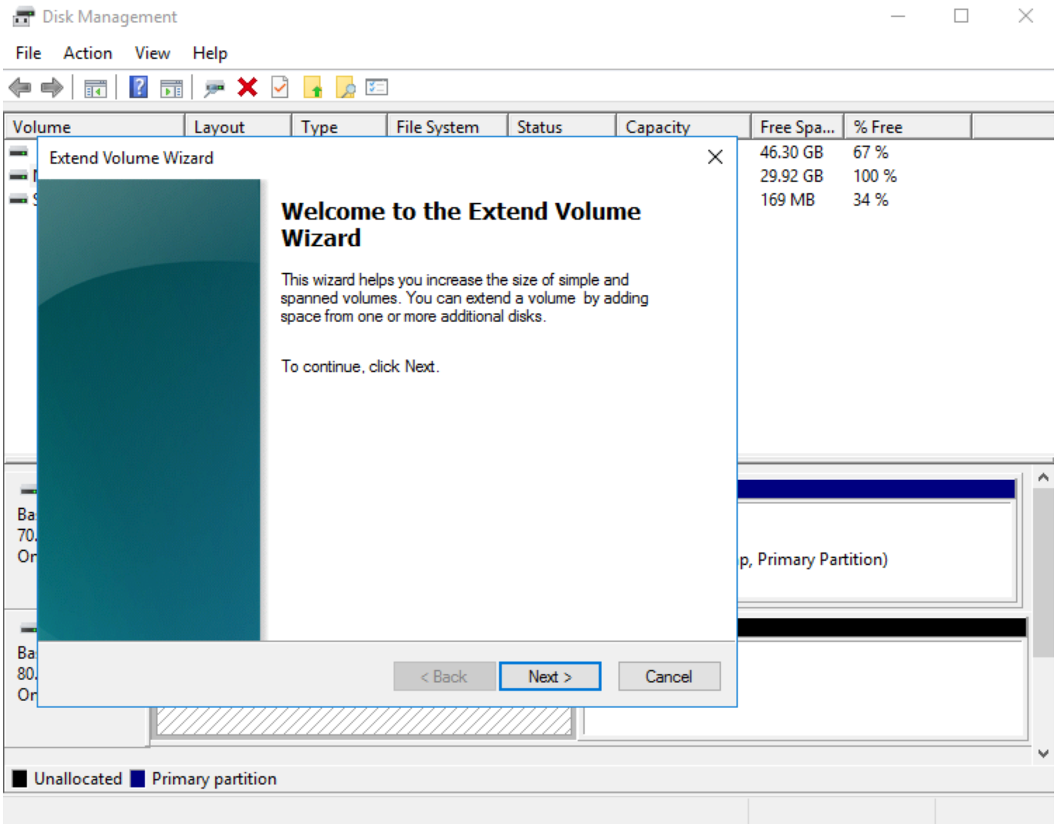
Step 3 Right-click the target volume and choose **Extend Volume**.

Figure 5-54 Choosing Extend Volume (Windows Server 2016 operating system)



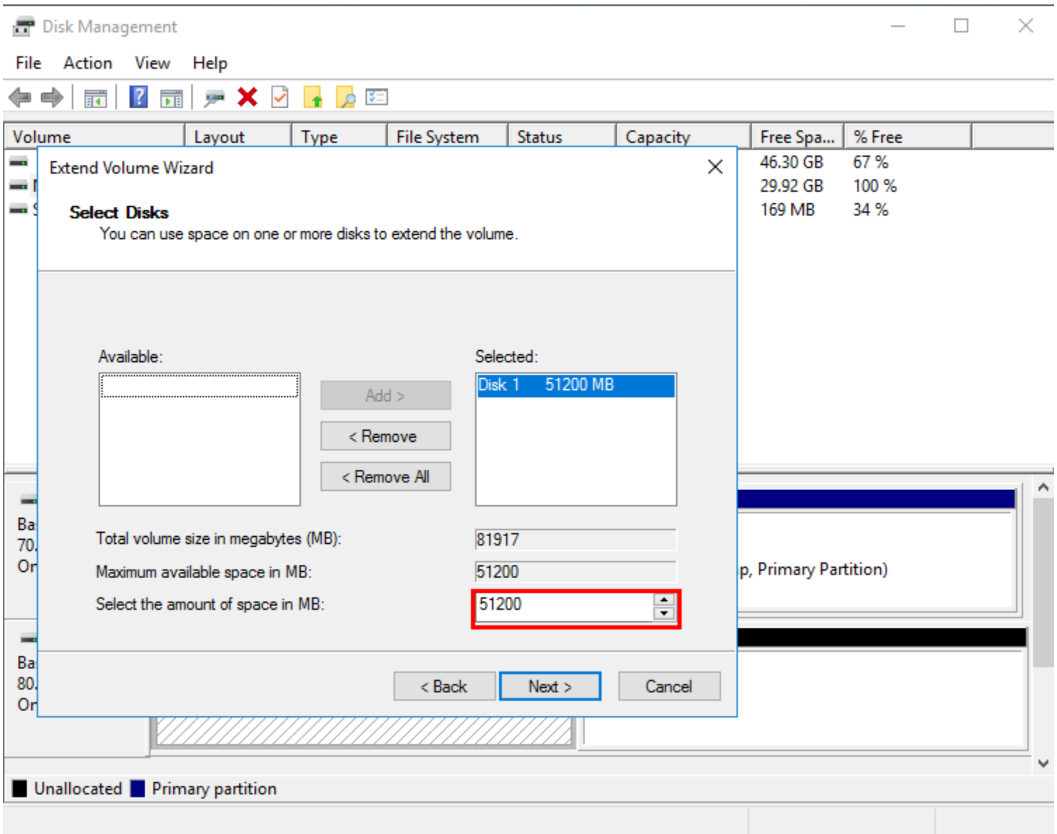
Step 4 On the displayed **Extend Volume Wizard** window, click **Next**.

Figure 5-55 Extend Volume Wizard (Windows Server 2016 operating system)



Step 5 In the text box to the right of **Select the amount of space in MB**, enter the amount of the additional space and click **Next**.

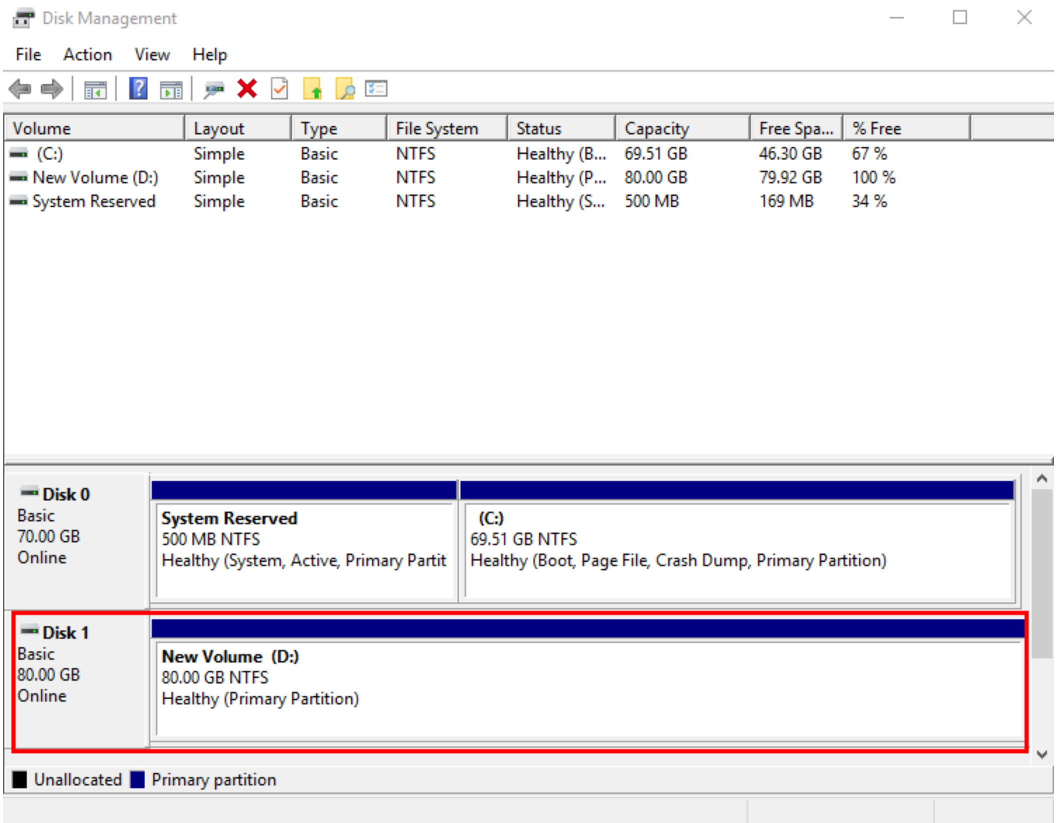
Figure 5-56 Selecting space (Windows Server 2016 operating system)



Step 6 Click **Finish**.

After the expansion succeeded, the partition size is larger than the original size.

Figure 5-57 Capacity expansion succeeded (Windows Server 2016 operating system)



----End

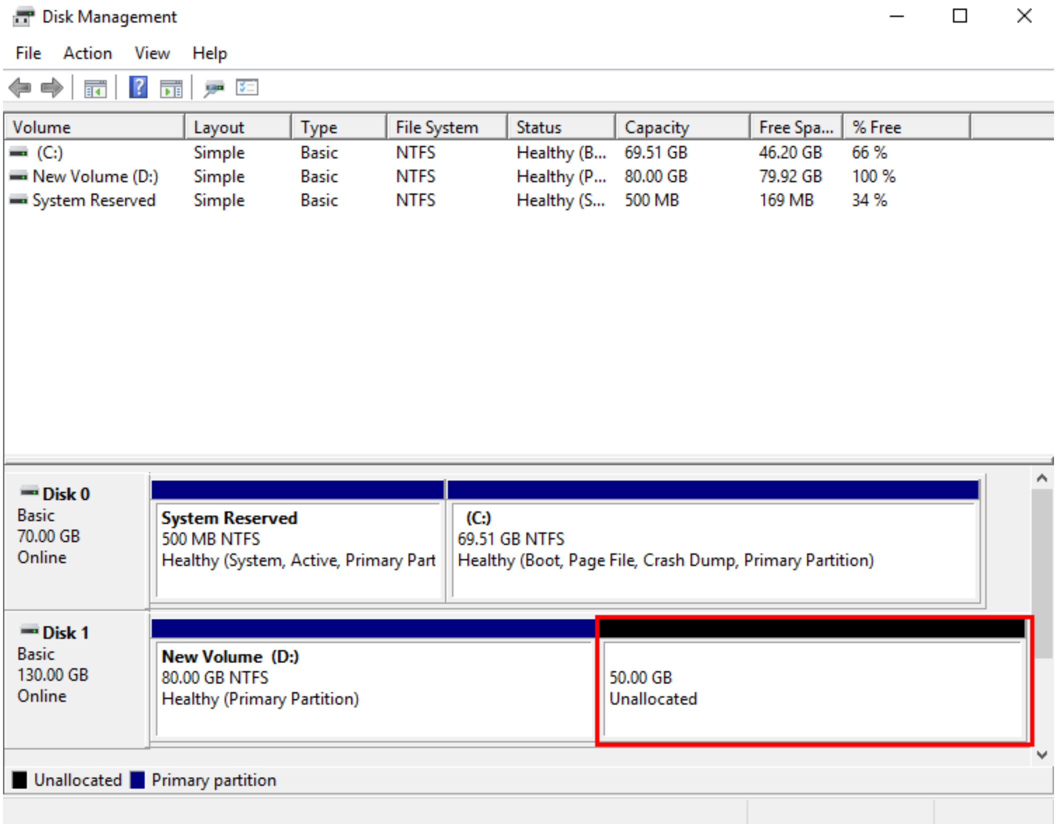
Data Disk: Create a New Volume with the Additional Space

In this example, the data disk has 80 GiB originally, and 50 GiB is added on the management console. The following procedure describes how to use this 50 GiB to create a new volume, for example volume (E:), on the server. After the operation is complete, new volume (E:) has 50 GiB of capacity and can be used as a data volume.

Step 1 On the desktop of the server, right-click the start icon in lower left corner and choose **Disk Management**.

The **Disk Management** window is displayed.

Figure 5-58 Unallocated disk space (Windows Server 2016 data disk)

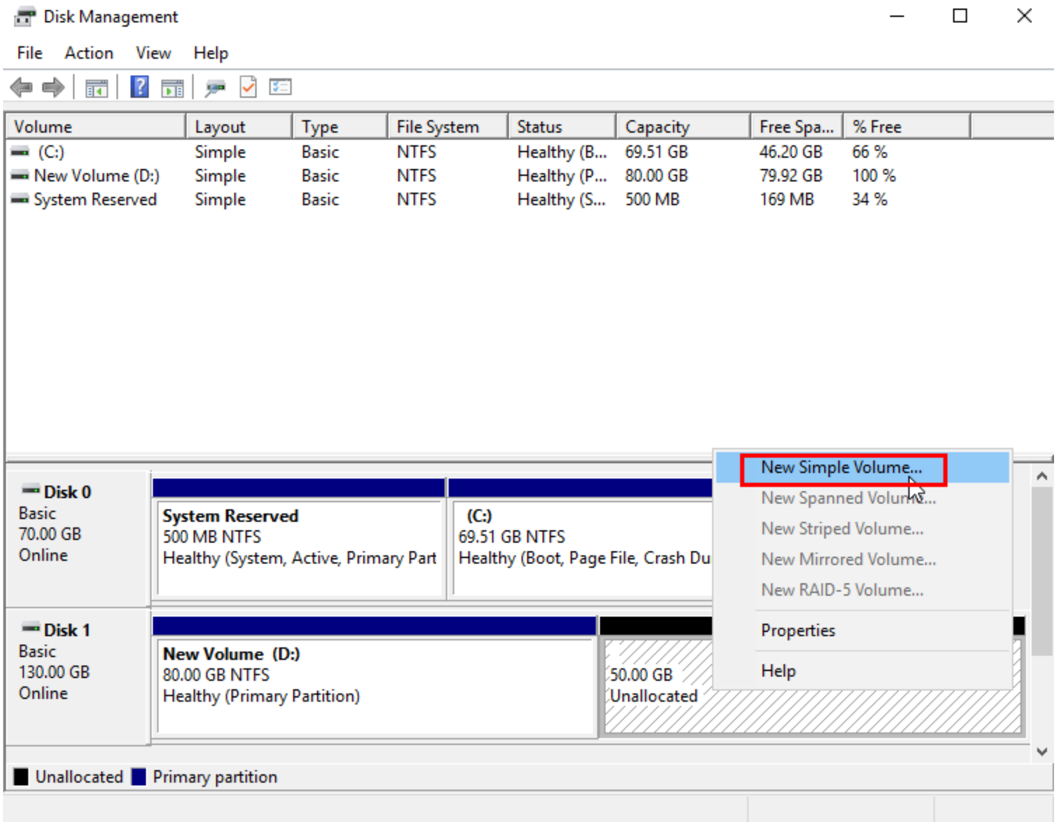


NOTE

If you cannot see the additional space, right-click **Disk Management** and choose **Refresh** from the shortcut menu.

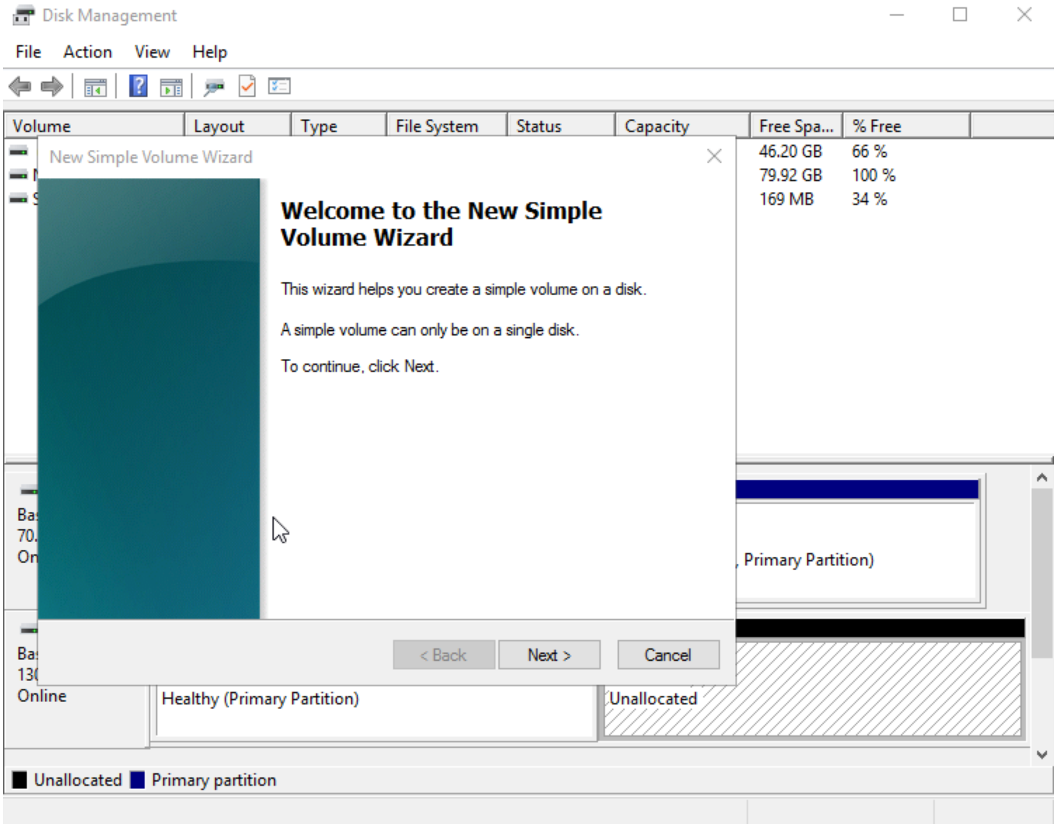
Step 2 In the **Unallocated** area of **Disk 1**, right-click the blank area and choose **New Simple Volume**.

Figure 5-59 New Simple Volume (Windows Server 2016 data disk)



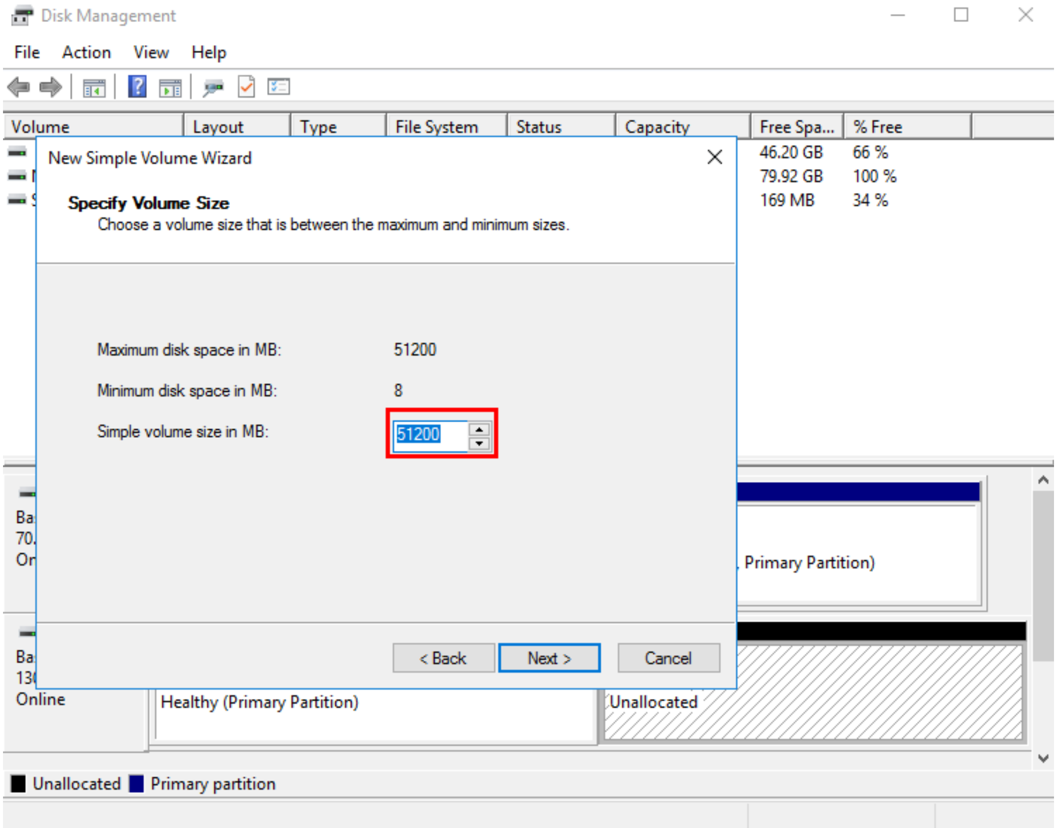
Step 3 On the displayed **New Simple Volume Wizard** window, click **Next**.

Figure 5-60 New Simple Volume Wizard (Windows Server 2016 data disk)



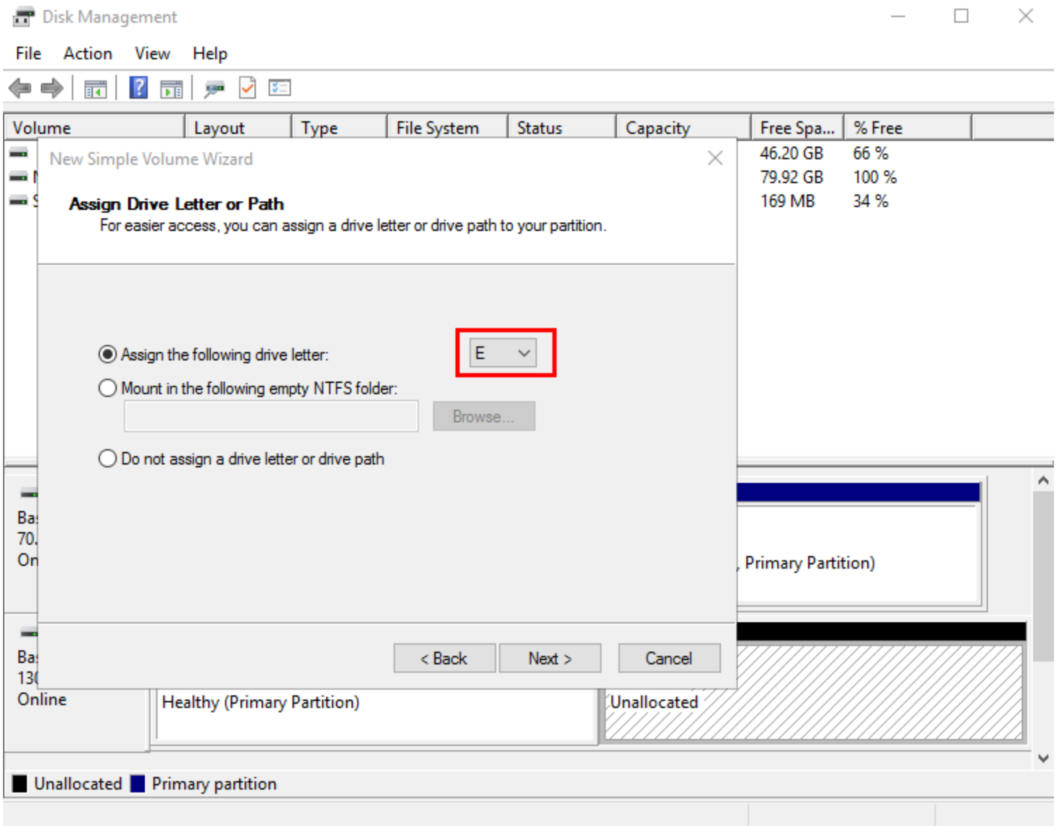
Step 4 On the displayed **Specify Volume Size** page, set **Simple volume size in MB** and click **Next**. In this example, the default size is used.

Figure 5-61 Specify Volume Size (Windows Server 2016 data disk)



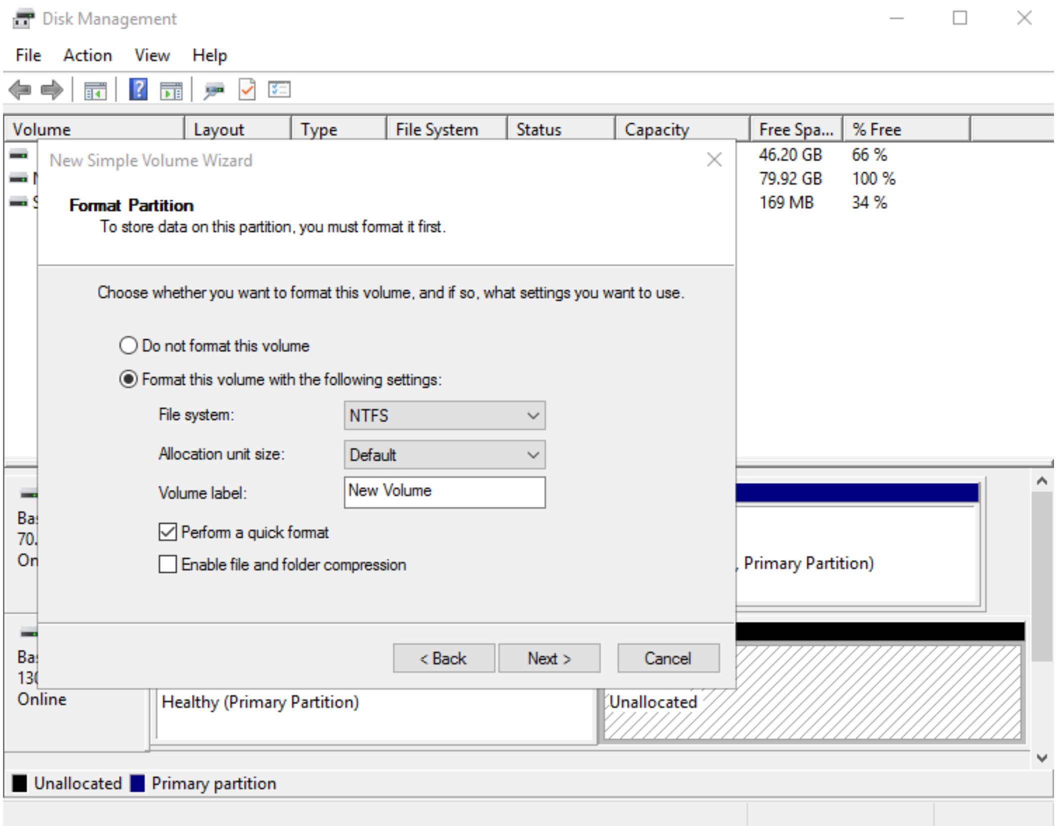
Step 5 On the displayed **Assign Drive Letter and Path** page, click **Assign the following drive letter**, select a drive letter, and click **Next**. In this example, drive letter **E** is selected.

Figure 5-62 Assign Drive Letter or Path (Windows Server 2016 data disk)



Step 6 On the displayed **Format Partition** page, click **Format this volume with the following settings**, set parameters based on the requirements, and select **Perform a quick format**. Then, click **Next**.

Figure 5-63 Format Partition (Windows Server 2016 data disk)



Step 7 Click **Finish**.

After the expansion succeeded, new volume (E:) is displayed.

Figure 5-64 Completed

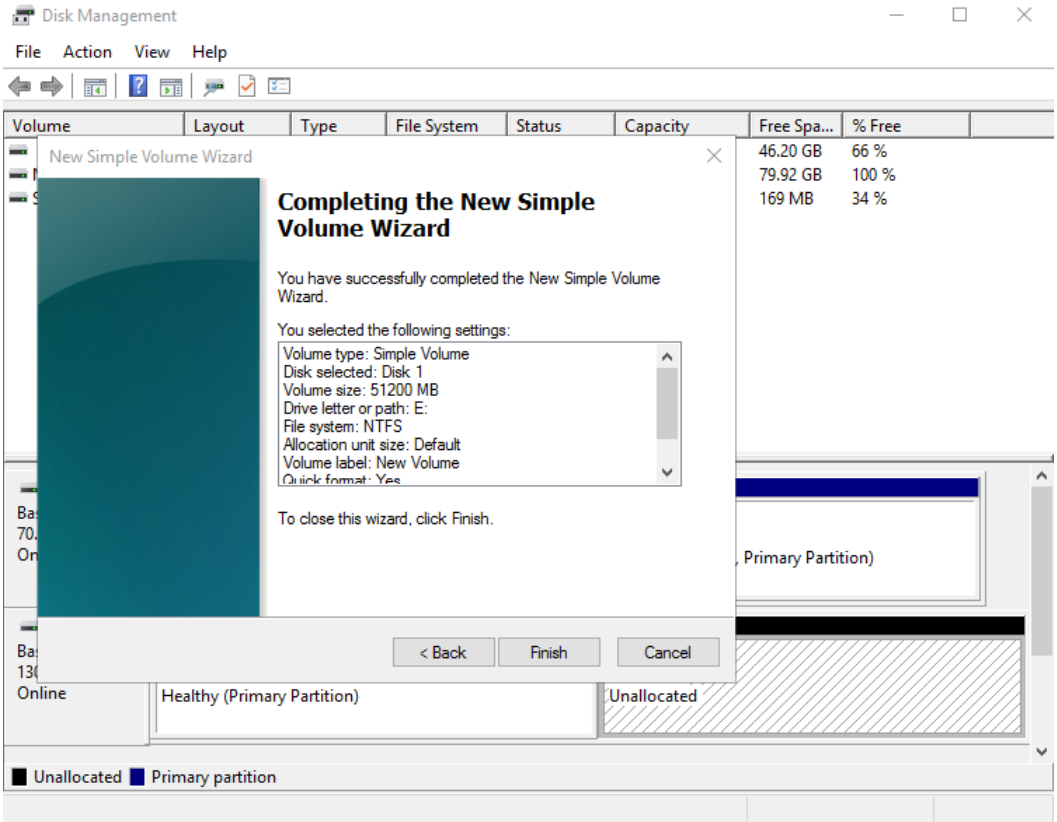
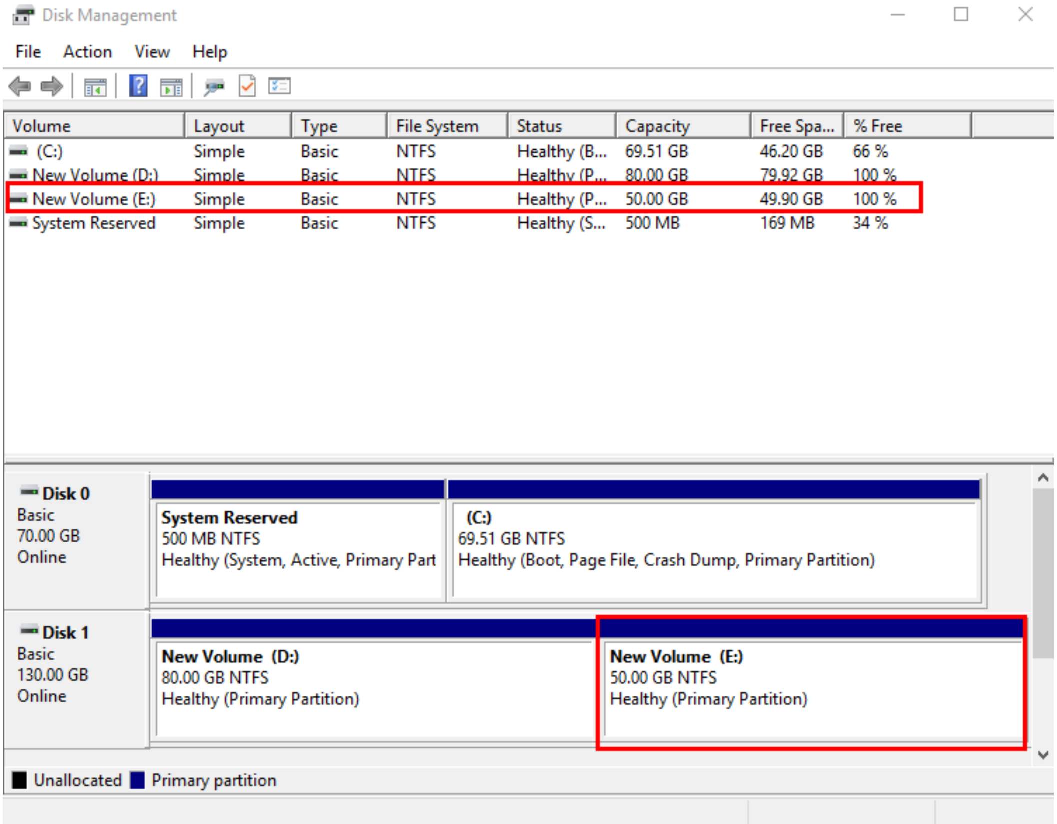


Figure 5-65 New Volume (E:)



----End

5.6 Extending Disk Partitions and File Systems (Linux)

5.6.1 Partition and File System Extension Preparations (Linux)

Before extending the disk partition and file system, you must check the disk partition style and file system format, and then select the appropriate operation accordingly.

1. To view the disk partition style, see the following methods:
   - **Method 1: Check Partition Style and File System Format Using fdisk**
   - **Method 2: Check Partition Style and File System Format Using parted**
2. To select the appropriate operation, see **Table 5-3**.

Table 5-3 Disk partition and file system extension scenarios

Disk	Scenario	Method
System disk	Create a new MBR partition with the additional space.	**Creating a New MBR Partition**

Disk	Scenario	Method
	Allocate the additional space to an existing MBR partition.	• **Extending an Existing MBR Partition (Kernel Version Later Than 3.6.0)** • **Extending an Existing MBR Partition (Kernel Version Earlier Than 3.6.0)**
Data disk	Create a new MBR partition with the additional space.	**Creating a New MBR Partition**
	Allocate the additional space to an existing MBR partition.	**Extending an Existing MBR Partition**
	Create a new GPT partition with the additional space.	**Creating a New GPT Partition**
	Allocate the additional space to an existing GPT partition.	**Extending an Existing GPT Partition**
SCSI data disk	Create a new MBR partition with the additional space.	**Creating a New MBR Partition**
	Allocate the additional space to an existing MBR partition.	**Extending an Existing MBR Partition**

NOTE

The maximum disk capacity that MBR supports is 2 TiB, and the disk space exceeding 2 TiB cannot be used.

If your disk uses MBR and you need to expand the disk capacity to over 2 TiB, change the partition style from MBR to GPT. Ensure that the disk data has been backed up before changing the partition style because services will be interrupted and data on the disk will be cleared during this change.

Method 1: Check Partition Style and File System Format Using fdisk

Step 1 Run the following command to view all the disks attached to the server:

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  150G  0 disk
└─vdb1 253:17   0  100G  0 part /mnt/sdc
```

In this example, data disk **/dev/vdb** already has partition **/dev/vdb1** before capacity expansion, and the additional 50 GiB added has not been allocated yet. Therefore, **/dev/vdb** has 150 GiB, and **/dev/vdb1** has 100 GiB.

NOTE

If you run **lsblk** and find out that disk **/dev/vdb** has no partitions, format the disk by referring to **How Do I Extend the File System of an Unpartitioned Data Disk in Linux?** and expand the capacity. Otherwise, the additional space cannot be used after expansion.

Step 2 Run the following command to view the current disk partition style:

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: 161.1 GiB, 161061273600 bytes, 314572800 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
```

The value in the **System** column indicates the disk partition style. Value **Linux** indicates the MBR partition style. Value **GPT** indicates the GPT partition style.

- If the disk partitions displayed are inconsistent with those obtained in **Step 1**, existing partitions uses GPT and there is unallocated disk space. In this case, you cannot query all the partitions using **fdisk -l**. Go to **Method 2: Check Partition Style and File System Format Using parted**.
- If the disk partitions displayed are consistent with those obtained in **Step 1**, continue with the following operations.

Step 3 Run the following command to view the partition's file system format:

blkid *Disk partition*

In this example, run the following command:

blkid /dev/vdb1

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

In the command output, the **TYPE** value is **ext4**, indicating that **/dev/vdb1**'s file system format is **ext4**.

Step 4 Run the following command to view the file system status:

ext*: **e2fsck -n** *Disk partition*

xfs: **xfs_repair -n** *Disk partition*

In this example, the ext4 file system is used. Therefore, run the following command:

e2fsck -n /dev/vdb1

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# e2fsck -n /dev/vdb1
e2fsck 1.42.9 (28-Dec-2013)
Warning!  /dev/vdb1 is mounted.
Warning: skipping journal recovery because doing a read-only filesystem check.
/dev/vdb1: clean, 11/6553600 files, 459544/26214144 blocks
```

If the file system status is **clean**, the file system is normal. Otherwise, rectify the faulty and then perform the capacity expansion.

----End

Method 2: Check Partition Style and File System Format Using parted

Step 1 Run the following command to view all the disks attached to the server:

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  150G  0 disk
└─vdb1 253:17   0  100G  0 part /mnt/sdc
```

In this example, data disk **/dev/vdb** already has partition **/dev/vdb1** before capacity expansion, and the additional 50 GiB added has not been allocated yet. Therefore, **/dev/vdb** has 150 GiB, and **/dev/vdb1** has 100 GiB.

NOTE

If you run **lsblk** and find out that disk **/dev/vdb** has no partitions, format the disk by referring to **How Do I Extend the File System of an Unpartitioned Data Disk in Linux?** and expand the capacity. Otherwise, the additional space cannot be used after expansion.

Step 2 Run the following command and enter **p** to view the disk partition style:

parted *Disk*

For example, run the following command to view **/dev/vdb**'s partition style:

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) p
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another
operating system believes the
disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of
the space (an extra 104857600
blocks) or continue with the current setting?
Fix/Ignore? Fix
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 161GiB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
```

```
Number  Start   End    Size   File system  Name  Flags
 1      1049kB  107GiB  107GiB  ext4         test

(parted)
```

Partition Table indicates the disk partition style. **Partition Table: msdos** means MBR, **Partition Table: gpt** means GPT, and **Partition Table: loop** means that the whole disk is partitioned.

- If the following error information is displayed, enter **Fix**.
```
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another operating system believes the
disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
```
  The GPT partition table information is stored at the start of the disk. To reduce the risk of damage, a backup of the information is saved at the end of the disk. When you expand the disk capacity, the end of the disk changes accordingly. In this case, enter **Fix** to move the backup file of the information to new disk end.
- If the following warning information is displayed, enter **Fix**.
```
Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of the space (an extra 104857600
blocks) or continue with the current setting?
Fix/Ignore? Fix
```
  Enter **Fix** as prompted. The system automatically sets the GPT partition style for the additional space.

Step 3 Enter **q** and press **Enter** to exit parted.

----End

5.6.2 Extending Partitions and File Systems for System Disks (Linux)

Scenarios

After a disk is expanded on the management console, the disk size is enlarged, but the additional space cannot be used directly.

In Linux, you must allocate the additional space to an existing partition or a new partition.

If the disk capacity is expanded when its server is stopped, the additional space of a Linux system disk will be automatically added to the partition at the disk end upon the server startup. In this case, the additional space can be used directly.

This section uses CentOS 7.4 64bit and CentOS 6.5 64bit as the sample OSs to describe how to extend the disk partition using growpart and fdisk. The method for allocating the additional space varies with the server OS. This section is used for reference only. For detailed operations and differences, see the corresponding OS documents.

For how to query the Linux kernel version, see **Querying the Linux Kernel Version**.

- **Extending an Existing MBR Partition (Kernel Version Later Than 3.6.0)**
- **Extending an Existing MBR Partition (Kernel Version Earlier Than 3.6.0)**

- **Creating a New MBR Partition**

NOTICE

Performing the expansion operations with caution. Incorrect operations may lead to data loss or exceptions, so you are advised to back up the disk data using CBR or snapshots before expansion. For details about using CBR, see **Managing EVS Backups**. For details about using snapshots, see the *Elastic Volume Service API Reference*

Prerequisites

- You have expanded the disk capacity and attached the disk to a server on the management console. For details, see **Expanding Capacity for an In-use EVS Disk** or **Expanding Capacity for an Available EVS Disk**.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Constraints

The additional space can only be added to the last partition of the disk.

Querying the Linux Kernel Version

Run the following command to query the Linux kernel version:

uname -a

Then, perform corresponding operations depending on whether the Linux kernel version is later than 3.6.0.

- For CentOS 7.4 64bit, information similar to the following is displayed:
  ```
  Linux ecs-test-0001 3.10.0-957.5.1.el7.x86_64 #1 SMP Fri Feb 1 14:54:57 UTC 2019 x86_64 x86_64
  ```
  The kernel version is 3.10.0, which is later than 3.6.0. For subsequent operations, see **Extending an Existing MBR Partition (Kernel Version Later Than 3.6.0)**.
- For CentOS 6.5 64bit, information similar to the following is displayed:
  ```
  Linux ecs-test-0002 2.6.32-754.10.1.el6.x86_64 #1 SMP Tue Jan 15 17:07:28 UTC 2019 x86_64
  ```
  The kernel version is 2.6.32, which is earlier than 3.6.0. In this case, the disk partition and file system extension take effect only after a server reboot. For subsequent operations, see **Extending an Existing MBR Partition (Kernel Version Earlier Than 3.6.0)**.

Extending an Existing MBR Partition (Kernel Version Later Than 3.6.0)

CentOS 7.4 64bit is used as the sample OS. Originally, system disk **/dev/vda** has 40 GiB and one partition (**/dev/vda1**), and then 60 GiB is added to the disk. The following procedure shows you how to allocate the additional 60 GiB to the existing MBR partition **/dev/vda1**.

Step 1 (Optional) Run the following command to install the growpart tool:

yum install cloud-utils-growpart

NOTE

You can run **growpart** to check whether the growpart tool has been installed. If the command output displays the tool usage instructions, the tool has been installed and you do not need to install it again.

Step 2 Run the following command to view the total capacity of the **/dev/vda** system disk:

fdisk -l

Information similar to the following is displayed:

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

Disk /dev/vda: 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: 0x000bcb4e

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

Step 3 Run the following command to view the capacity of the **/dev/vda1** partition:

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  2.0G   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
```

Step 4 Run the following command to extend the partition using growpart:

growpart *System disk Partition number*

In this example, run the following command:

growpart /dev/vda 1

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# growpart /dev/vda 1
CHANGED: partition=1 start=2048 old: size=83884032 end=83886080 new: size=209713119,end=209715167
```

Step 5 Run the following command to extend the file system of the partition:

resize2fs *Disk partition*

In this example, run the following command:

resize2fs /dev/vda1

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# resize2fs /dev/vda1
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/vda1 is mounted on /; on-line resizing required
old_desc_blocks = 5, new_desc_blocks = 13
The filesystem on /dev/vda1 is now 26214139 blocks long.
```

NOTE

If the error message "open: No such file or directory while opening /dev/vdb1" is returned, an incorrect partition is specified. Run **df -TH** to view the disk partitions.

Step 6 Run the following command to view the new capacity of the **/dev/vda1** partition:

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      106G  2.0G   99G   2% /
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
```

----End

Extending an Existing MBR Partition (Kernel Version Earlier Than 3.6.0)

NOTICE

If the OS kernel version is earlier than 3.6.0, the extension of an existing MBR disk partition and file system takes effect only after a server reboot, and services will be interrupted.

CentOS 6.5 64bit is used as the sample OS. Originally, system disk **/dev/vda** has 40 GiB and one partition (**/dev/vda1**), and then 60 GiB is added to the disk. The following procedure shows you how to allocate the additional 60 GiB to the existing MBR partition **/dev/vda1**.

Step 1 (Optional) Run the following command to install the growpart tool:

yum install cloud-utils-growpart

NOTE

You can run **growpart** to check whether the growpart tool has been installed. If the command output displays the tool usage instructions, the tool has been installed and you do not need to install it again.

Step 2 Run the following command to install the dracut-modules-growroot tool:

yum install cloud-utils-growpart

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# yum install cloud-utils-growpart
Loaded plugins: fastestmirror, security
Setting up Install Process
Determining fastest mirrors
...
```

```
Package cloud-utils-growpart-0.27-10.el6.x86_64 already installed and latest version
Nothing to do
```

Step 3 Run the following command to regenerate the initramfs file:

dracut -f

Step 4 Run the following command to view the total capacity of the **/dev/vda** system disk:

fdisk -l

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# fdisk -l

Disk /dev/vda: 107.4 GiB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0004e0be

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *           1        5222    41942016   83  Linux
```

Step 5 Run the following command to view the capacity of the **/dev/vda1** partition:

df -TH

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# df -TH
Filesystem     Type   Size  Used Avail Use% Mounted on
/dev/vda1      ext4    43G  1.7G   39G   5% /
tmpfs          tmpfs  2.1G     0  2.1G   0% /dev/shm
```

Step 6 Run the following command to extend the partition using growpart:

growpart *System disk Partition number*

In this example, run the following command:

growpart /dev/vda 1

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# growpart /dev/vda 1
CHANGED: partition=1 start=2048 old: size=83884032 end=83886080 new: size=209710462,end=209712510
```

Step 7 Run the following command to restart the server:

reboot

After the server is restarted, reconnect to the server and perform the following steps.

Step 8 Run the following command to extend the file system of the partition:

resize2fs *Disk partition*

In this example, run the following command:

resize2fs /dev/vda1

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# resize2fs /dev/vda1
resize2fs 1.41.12 (17-May-2010)
The filesystem is already 26213807 blocks long.  Nothing to do!
```


NOTE

If the error message "open: No such file or directory while opening /dev/vdb1" is returned, an incorrect partition is specified. Run **df -TH** to view the disk partitions.

Step 9 Run the following command to view the new capacity of the **/dev/vda1** partition:

df -TH

Information similar to the following is displayed:

```
[root@ecs-test-0002 ~]# df -TH
Filesystem     Type   Size  Used Avail Use% Mounted on
/dev/vda1      ext4   106G  1.7G   99G   2% /
tmpfs         tmpfs  2.1G     0  2.1G   0% /dev/shm
```

----End

Creating a New MBR Partition

Originally, system disk **/dev/vda** has 40 GiB and one partition (**/dev/vda1**), and then 40 GiB is added to the disk. The following procedure shows you how to create a new MBR partition **/dev/vda2** with this 40 GiB.

Step 1 Run the following command to view the disk partition information:

fdisk -l

Information similar to the following is displayed:

```
[root@ecs-2220 ~]# fdisk -l

Disk /dev/vda: 85.9 GiB, 85899345920 bytes, 167772160 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: 0x0008d18f

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

In the command output, the capacity of the **dev/vda** system disk is 80 GiB, in which the in-use **dev/vda1** partition takes 40 GiB and the additional 40 GiB has not been allocated.

Step 2 Run the following command to enter fdisk:

fdisk /dev/vda

Information similar to the following is displayed:

```
[root@ecs-2220 ~]# fdisk /dev/vda
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.


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 (1 primary, 0 extended, 3 free)
   e   extended
```

There are two types of disk partitions:

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

NOTE

If MBR is used, a maximum of 4 primary partitions, or 3 primary partitions plus 1 extended partition can be created. The extended partition must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

Step 4 In this example, a primary partition is created. Therefore, enter **p** and press **Enter** to create a primary partition.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (2-4, default 2):
```

Step 5 Enter the serial number of the primary partition and press **Enter**. Partition number **2** is used in this example. Therefore, enter **2** and press **Enter.**

Information similar to the following is displayed:

```
Partition number (2-4, default 2): 2
First sector (83886080-167772159, default 83886080):
```

Step 6 Enter the new partition's start sector and press **Enter**. In this example, the default start sector is used.

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 (83886080-167772159, default 83886080):
Using default value 83886080
Last sector, +sectors or +size{K,M,G} (83886080-167772159,default 167772159):
```

Step 7 Enter the new partition's end sector and press **Enter**. In this example, the default end sector is used.

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} (83886080-167772159,
default 167772159):
Using default value 167772159
Partition 2 of type Linux and of size 40 GiB is set

Command (m for help):
```

Step 8 Enter **p** and press **Enter** to view the new partition.

Information similar to the following is displayed:

```
Command (m for help): p

Disk /dev/vda: 85.9 GiB, 85899345920 bytes, 167772160 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: 0x0008d18f
```

```
  Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *        2048    83886079    41942016   83  Linux
/dev/vda2        83886080   167772159    41943040   83  Linux
Command (m for help):
```

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.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
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 Run the following command to synchronize the new partition table to the OS:

partprobe

Step 11 Run the following command to set the file system format for the new partition:

mkfs -t *File system Disk partition*

- Sample command of the ext* file system:

  (The ext4 file system is used in this example.)

  mkfs -t ext4 /dev/vda2

  Information similar to the following is displayed:

```
[root@ecs-2220 ~]# mkfs -t ext4 /dev/vda2
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
2621440 inodes, 10485760 blocks
524288 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2157969408
320 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

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

- Sample command of the xfs file system:

  mkfs -t xfs /dev/vda2

  Information similar to the following is displayed:

```
[root@ecs-2220 ~]# mkfs -t xfs /dev/vda2
meta-data=/dev/vda2              isize=512    agcount=4, agsize=2621440 blks
```

```
         =                       sectsz=512    attr=2, projid32bit=1
         =                       crc=1         finobt=0, sparse=0
data     =                       bsize=4096    blocks=10485760, imaxpct=25
         =                       sunit=0       swidth=0 blks
naming   =version2               bsize=4096    ascii-ci=0 ftype=1
log      =internal log           bsize=4096    blocks=5120, version=2
         =                       sectsz=512    sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096    blocks=0, rtextents=0
```

The formatting takes a while, and you need to observe the system running status. Once **done** is displayed in the command output, the formatting is complete.

Step 12 (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/opt** mount point:

mkdir /opt

Step 13 Run the following command to mount the new partition:

mount *Disk partition Mount point*

In this example, run the following command to mount the new partition **/dev/vda2** on **/opt**:

mount /dev/vda2 /opt

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

Step 14 Run the following command to view the mount result:

df -TH

Information similar to the following is displayed:

```
[root@ecs-2220 ~]# df -TH
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/vda1      ext4       43G  2.0G   39G   5% /
devtmpfs       devtmpfs  509M     0  509M   0% /dev
tmpfs          tmpfs     520M     0  520M   0% /dev/shm
tmpfs          tmpfs     520M  7.2M  513M   2% /run
tmpfs          tmpfs     520M     0  520M   0% /sys/fs/cgroup
tmpfs          tmpfs     104M     0  104M   0% /run/user/0
/dev/vda2      ext4       43G   51M   40G   1% /opt
```

NOTE

If the server is restarted, the mounting will become invalid. You can modify the **/etc/fstab** file to configure automount at startup. See the following part for details.

----End

Configuring Automatic Mounting at System Start

The **fstab** file controls what disks are automatically mounted at server startup. You can configure the **fstab** file of a server that has data. This operation will not affect the existing data.

The following example uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names (like **/dev/vdb1**) 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 server stop or start. This can even prevent your server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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

5.6.3 Extending Partitions and File Systems for Data Disks (Linux)

Scenarios

After a disk is expanded on the management console, the disk size is enlarged, but the additional space cannot be used directly.

In Linux, you must allocate the additional space to an existing partition or a new partition.

This section uses CentOS 7.4 64bit as the sample OS to describe how to extend an MBR or GPT partition. The method for allocating the additional space varies with the server OS. This section is used for reference only. For detailed operations and differences, see the corresponding OS documents.

- **Creating a New MBR Partition**
- **Extending an Existing MBR Partition**
- **Creating a New GPT Partition**
- **Extending an Existing GPT Partition**

NOTICE

Performing the expansion operations with caution. Incorrect operations may lead to data loss or exceptions, so you are advised to back up the disk data using CBR or snapshots before expansion. For details about using CBR, see **Managing EVS Backups**. For details about using snapshots, see the *Elastic Volume Service API Reference*

Prerequisites

- You have expanded the disk capacity and attached the disk to a server on the management console. For details, see **Expanding Capacity for an In-use EVS Disk** or **Expanding Capacity for an Available EVS Disk**.
- You have logged in to the server.
  – For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  – For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Constraints

- The additional space of a data disk cannot be added to the root partition. To extend the root partition, expand the system disk instead.
- The additional space can only be added to the last partition of the disk.

Creating a New MBR Partition

Originally, data disk **/dev/vdb** has 100 GiB and one partition (**/dev/vdb1**), and then 50 GiB is added to the disk. The following procedure shows you how to create a new MBR partition **/dev/vdb2** with this 50 GiB.

Step 1 Run the following command to view the disk partition information:

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: 161.1 GiB, 161061273600 bytes, 314572800 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
```

Step 2 Run the following command to enter fdisk:

fdisk *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.


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 (1 primary, 0 extended, 3 free)
   e   extended
Select (default p):
```

There are two types of disk partitions:

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

NOTE

If MBR is used, a maximum of 4 primary partitions, or 3 primary partitions plus 1 extended partition can be created. The extended partition must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

Step 4 In this example, a primary partition is created. Therefore, enter **p** and press **Enter** to create a primary partition.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (2-4, default 2):
```

Partition number indicates the serial number of the primary partition. Because partition number 1 has been used, the value ranges from **2** to **4**.

Step 5 Enter the serial number of the primary partition and press **Enter**. Partition number **2** is used in this example. Therefore, enter **2** and press **Enter.**

Information similar to the following is displayed:

```
Partition number (2-4, default 2): 2
First sector (209715200-314572799, default 209715200):
```

First sector indicates the start sector. The value ranges from **209715200** to **314572799**, and the default value is **209715200**.

Step 6 Enter the new partition's start sector and press **Enter**. In this example, the default start sector is used.

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 (209715200-314572799, default 209715200):
Using default value 209715200
Last sector, +sectors or +size{K,M,G} (209715200-314572799, default 314572799):
```

Last sector indicates the end sector. The value ranges from **209715200** to **314572799**, and the default value is **314572799**.

Step 7 Enter the new partition's end sector and press **Enter**. In this example, the default end sector is used.

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} (209715200-314572799, default 314572799):
Using default value 314572799
Partition 2 of type Linux and of size 50 GiB is set

Command (m for help):
```

Step 8 Enter **p** and press **Enter** to view the new partition.

Information similar to the following is displayed:

```
Command (m for help): p

Disk /dev/vdb: 161.1 GiB, 161061273600 bytes, 314572800 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
/dev/vdb2       209715200   314572799    52428800   83  Linux

Command (m for help):
```

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.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

NOTE

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

Step 10 Run the following command to synchronize the new partition table to the OS:

partprobe

Step 11 Run the following command to set the file system format for the new partition:

mkfs -t *File system Disk partition*

- Sample command of the ext* file system:

  mkfs -t ext4 /dev/vdb2

  Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# mkfs -t ext4 /dev/vdb2
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
3276800 inodes, 13107200 blocks
655360 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2162163712
400 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

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

- Sample command of the xfs file system:

  mkfs -t xfs /dev/vdb2

  Information similar to the following is displayed:


```
[root@ecs-test-0001 ~]# mkfs -t xfs /dev/vdb2
meta-data=/dev/vdb2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=6400, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

The formatting takes a while, and you need to observe the system running status. Once **done** is displayed in the command output, the formatting is complete.

Step 12 (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

Step 13 Run the following command to mount the new partition:

mount *Disk partition Mount point*

In this example, run the following command to mount the new partition **/dev/vdb2** on **/mnt/test**:

mount /dev/vdb2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

Step 14 Run the following command to view 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
/dev/vdb2      ext4       53G   55M   50G   1% /mnt/test
```

NOTE

If the server is restarted, the mounting will become invalid. You can modify the **/etc/fstab** file to configure automount at startup. For details, see **Configuring Automatic Mounting at System Start**.

----End

Extending an Existing MBR Partition

NOTICE

If the additional space is allocated to an existing partition, data on the disk will not be cleared but you must use **umount** to unmount the existing partition. In this case, services will be affected.

Originally, data disk **/dev/vdb** has 150 GiB and two partitions (**/dev/vdb1** and **/dev/vdb2**), and then 80 GiB is added to the disk. The following procedure shows you how to add this 80 GiB to the existing MBR partition **/dev/vdb2**.

NOTICE

During an expansion, the additional space is added to the end of the disk. Therefore, if the disk has multiple partitions, the additional space can only be allocated to the partition at the disk end.

Step 1 Run the following command to view the disk partition information:

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: 247.0 GiB, 246960619520 bytes, 482344960 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
/dev/vdb2       209715200   314572799    52428800   83  Linux
```

In the command output, take note of the partition's start and end sectors. In this example, **/dev/vdb2**'s start sector is **209715200**, and its end sector is **314572799**.

View the **/dev/vdb** capacity and check whether the additional space is included.

- If the additional space is not included, refresh the capacity according to **Extending Partitions and File Systems for SCSI Disks (Linux)**.
- If the additional space is included, take note of the start and end sectors of the target partition and then go to **Step 2**. These values will be used in the subsequent operations.

Step 2 Run the following command to unmount the partition:

umount *Disk partition*

In this example, run the following command:

umount /dev/vdb2

Step 3 Run the following command to enter fdisk:

fdisk *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.


Command (m for help):
```

Step 4 Run the following command to delete the partition to be extended:

1. Enter **d** and press **Enter** to delete the partition.

   Information similar to the following is displayed:

   ```
   Command (m for help): d
   Partition number (1,2, default 2):
   ```

2. Enter the partition number and press **Enter** to delete the partition. In this example, enter **2**.

   Information similar to the following is displayed:

   ```
   Partition number (1,2, default 2): 2
   Partition 2 is deleted

   Command (m for help):
   ```

   NOTE

   After deleting the partition, re-create the partition according to the following steps, and data on this disk will not be lost.

Step 5 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 (1 primary, 0 extended, 3 free)
   e   extended
Select (default p):
```

There are two types of disk partitions:

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

NOTE

If MBR is used, a maximum of 4 primary partitions, or 3 primary partitions plus 1 extended partition can be created. The extended partition must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

Step 6 Ensure that the entered partition type is the same as the partition had before. In this example, a primary partition is used. Therefore, enter **p** and press **Enter** to create a primary partition.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (2-4, default 2):
```

Partition number indicates the serial number of the primary partition.

Step 7 Ensure that entered partition number is the same as the partition had before. In this example, partition number **2** is used. Therefore, enter **2** and press **Enter**.

Information similar to the following is displayed:

```
Partition number (2-4, default 2): 2
First sector (209715200-482344959, default 209715200):
```

In the command output, **First sector** specifies the start sector.

NOTE

Data will be lost if the following operations are performed:

- Select a start sector other than the partition had before.
- Select an end sector smaller than the partition had before.

Step 8 Ensure that the entered start sector is the same as the partition had before. In this example, start sector **209715200** is recorded in **Step 1**. Therefore, enter **209715200** and press **Enter**.

Information similar to the following is displayed:

```
First sector (209715200-482344959, default 209715200):
Using default value 209715200
Last sector, +sectors or +size{K,M,G} (209715200-482344959, default 482344959):
```

In the command output, **Last sector** specifies the end sector.

Step 9 Ensure that the entered end sector is greater than or equal to the end sector recorded in **Step 1**. In this example, the recorded end sector is **314572799**, and the default end sector is used. Therefore, enter **482344959** and press **Enter**.

Information similar to the following is displayed:

```
Using default value 209715200
Last sector, +sectors or +size{K,M,G} (209715200-482344959, default 482344959):
Using default value 482344959
Partition 2 of type Linux and of size 130 GiB is set

Command (m for help):
```

The partition is created.

Step 10 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: 247.0 GiB, 246960619520 bytes, 482344960 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
/dev/vdb2       209715200   482344959   136314880   83  Linux

Command (m for help):
```

Step 11 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.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

NOTE

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

Step 12 Run the following command to synchronize the new partition table to the OS:

partprobe

Step 13 Perform the following operations based on the file system of the disk:

- For the **ext*** file system
  a. Run the following command to check the correctness of the file system on the partition:

     e2fsck -f *Disk partition*

     In this example, run the following command:

     e2fsck -f /dev/vdb2

     Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# e2fsck -f /dev/vdb2
e2fsck 1.42.9 (28-Dec-2013)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/vdb2: 11/3276800 files (0.0% non-contiguous), 251790/13107200 blocks
```

  b. Run the following command to extend the file system of the partition:

     resize2fs *Disk partition*

     In this example, run the following command:

     resize2fs /dev/vdb2

     Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# resize2fs /dev/vdb2
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/vdb2 to 34078720 (4k) blocks.
The filesystem on /dev/vdb2 is now 34078720 blocks long.
```

NOTE

If the error message "open: No such file or directory while opening /dev/vdb1" is returned, an incorrect partition is specified. Run **df -TH** to view the disk partitions.

c. (Optional) Run the following command to create a mount point:

   Perform this step if you want to mount the partition on a new mount point.

   mkdir *Mount point*

   In this example, run the following command to create the **/mnt/test** mount point:

   mkdir /mnt/test

d. Run the following command to mount the partition:

   mount *Disk partition Mount point*

   In this example, run the following command to mount partition **/dev/vdb2** on **/mnt/test**:

   mount /dev/vdb2 /mnt/test

   NOTE

   If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

- For the **xfs** file system

  a. (Optional) Run the following command to create a mount point:

     Perform this step if you want to mount the partition on a new mount point.

     mkdir *Mount point*

     In this example, run the following command to create the **/mnt/test** mount point:

     mkdir /mnt/test

  b. Run the following command to mount the partition:

     mount *Disk partition Mount point*

     In this example, run the following command to mount partition **/dev/vdb2** on **/mnt/test**:

     mount /dev/vdb2 /mnt/test

     NOTE

     If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

  c. Run the following command to extend the file system of the partition:

     sudo xfs_growfs *Disk partition*

In this example, run the following command:

sudo xfs_growfs /dev/vdb2

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# sudo xfs_growfs /dev/vdb2
meta-data=/dev/vdb2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, spinodes=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal               bsize=4096   blocks=6400, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 13107200 to 34078720.
```

Step 14 Run the following command to view 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
/dev/vdb2      ext4      138G   63M  131G   1% /mnt/test
```

----End

Creating a New GPT Partition

Originally, data disk **/dev/vdb** has 100 GiB and one partition (**/dev/vdb1**), and then 50 GiB is added to the disk. The following procedure shows you how to create a new GPT partition **/dev/vdb2** with this 50 GiB.

Step 1 Run the following command to 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  150G  0 disk
└─vdb1 253:17   0  100G  0 part /mnt/sdc
```

Step 2 Run the following command to enter parted:

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

Step 4 Enter **p** and press **Enter** to view the disk partition information.

Information similar to the following is displayed:

```
(parted) unit s
(parted) p
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another
operating system believes the
disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of
the space (an extra 104857600
blocks) or continue with the current setting?
Fix/Ignore? Fix
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 314572800s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start  End         Size        File system  Name  Flags
 1      2048s  209713151s  209711104s  ext4         test

(parted)
```

In the command output, take note of the partition's end sector. In this example, the end sector of the **/dev/vdb1** partition is **209713151s**.

- If the following error information is displayed, enter **Fix**.

  ```
  Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that
  another operating system believes the
  disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
  ```

  The GPT partition table information is stored at the start of the disk. To reduce the risk of damage, a backup of the information is saved at the end of the disk. When you expand the disk capacity, the end of the disk changes accordingly. In this case, enter **Fix** to move the backup file of the information to new disk end.

- If the following warning information is displayed, enter **Fix**.

  ```
  Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all
  of the space (an extra 104857600
  blocks) or continue with the current setting?
  Fix/Ignore? Fix
  ```

  Enter **Fix** as prompted. The system automatically sets the GPT partition style for the additional space.

Step 5 Run the following command and press **Enter**:

mkpart *Partition name Start sector End sector*

In this example, run the following command:

mkpart data 209713152s 100%

In this example, the additional space is used to create a new partition. In **Step 4**, the end sector of partition **dev/vdb1** is **209713151s**. Therefore, the start sector of the new partition **dev/vdb2** is set to **209713152s** and the end sector **100%**. This start and end sectors are for reference only. You can plan the number of partitions and partition size based on service requirements.

Information similar to the following is displayed:


```
(parted) mkpart data 209713152s 100%
(parted)
```

NOTE

The maximum sector can be obtained in either of the following ways:

- Query the disk's maximum end sector. For details, see **Step 2** to **Step 4**.
- Enter **-1s** or **100%**, and the value displayed is the maximum end sector.

Step 6 Enter **p** and press **Enter** to view the new partition.

Information similar to the following is displayed:

```
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 314572800s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start       End         Size        File system  Name  Flags
 1      2048s       209713151s  209711104s  ext4         test
 2      209713152s  314570751s  104857600s               data

(parted)
```

Step 7 Run the following command to set the file system format for the new partition:

mkfs -t *File system Disk partition*

- Sample command of the ext* file system:

  mkfs -t ext4 /dev/vdb2

  Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# mkfs -t ext4 /dev/vdb2
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
3276800 inodes, 13107200 blocks
655360 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2162163712
400 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

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

- Sample command of the xfs file system:

  mkfs -t xfs /dev/vdb2

  Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# mkfs -t xfs /dev/vdb2
meta-data=/dev/vdb2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
```



```
naming   =version2              bsize=4096    ascii-ci=0 ftype=1
log      =internal log          bsize=4096    blocks=6400, version=2
         =                      sectsz=512    sunit=0 blks, lazy-count=1
realtime =none                  extsz=4096    blocks=0, rtextents=0
```

The formatting takes a while, and you need to observe the system running status. Once **done** is displayed in the command output, the formatting is complete.

Step 8 (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

Step 9 Run the following command to mount the new partition:

mount *Disk partition Mount point*

In this example, run the following command to mount the new partition **/dev/vdb2** on **/mnt/test**:

mount /dev/vdb2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

Step 10 Run the following command to view 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
/dev/vdb2      ext4       53G   55M   50G   1% /mnt/test
```

NOTE

If the server is restarted, the mounting will become invalid. You can modify the **/etc/fstab** file to configure automount at startup. For details, see **Configuring Automatic Mounting at System Start**.

----End

Extending an Existing GPT Partition

NOTICE

If the additional space is allocated to an existing partition, data on the disk will not be cleared but you must use **umount** to unmount the existing partition. In this case, services will be affected.

Originally, data disk **/dev/vdb** has 150 GiB and two partitions (**/dev/vdb1** and **/dev/vdb2**), and then 80 GiB is added to the disk. The following procedure shows you how to add this 80 GiB to the existing GPT partition **/dev/vdb2**.

During an expansion, the additional space is added to the end of the disk. Therefore, if the disk has multiple partitions, the additional space can only be allocated to the partition at the disk end.

Step 1 Run the following command to 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  230G  0 disk
├─vdb1 253:17   0  100G  0 part /mnt/sdc
└─vdb2 253:18   0   50G  0 part /mnt/test
```

View the **/dev/vdb** capacity and check whether the additional space is included.

- If the additional space is not included, refresh the capacity according to **Extending Partitions and File Systems for SCSI Disks (Linux)**.
- If the additional space is included, go to **Step 2**.

Step 2 Run the following command to unmount the partition:

umount *Disk partition*

In this example, run the following command:

umount /dev/vdb2

Step 3 Run the following command to view the unmount result:

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  230G  0 disk
├─vdb1 253:17   0  100G  0 part /mnt/sdc
└─vdb2 253:18   0   50G  0 part
```

Step 4 Run the following command to enter parted:

parted *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 5 Enter **unit s** and press **Enter** to set the measurement unit of the disk to sector.

Step 6 Enter **p** and press **Enter** to view the disk partition information.

Information similar to the following is displayed:

```
(parted) unit s
(parted) p
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another operating system believes the
disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of the space (an extra 167772160
blocks) or continue with the current setting?
Fix/Ignore? Fix
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 482344960s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start       End         Size        File system  Name  Flags
 1      2048s       209713151s  209711104s  ext4         test
 2      209713152s  314570751s  104857600s  ext4         data

(parted)
```

Take note of the start and end sectors of the **/dev/vdb2** partition. These values will be used during the partition recreation. In this example, the partition's start sector is **209713152s**, and its end sector is **314570751s**.

- If the following error information is displayed, enter **Fix**.

  ```
  Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another operating system believes the
  disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
  ```

  The GPT partition table information is stored at the start of the disk. To reduce the risk of damage, a backup of the information is saved at the end of the disk. When you expand the disk capacity, the end of the disk changes accordingly. In this case, enter **Fix** to move the backup file of the information to new disk end.

- If the following warning information is displayed, enter **Fix**.

  ```
  Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of the space (an extra 104857600
  blocks) or continue with the current setting?
  Fix/Ignore? Fix
  ```

  Enter **Fix** as prompted. The system automatically sets the GPT partition style for the additional space.

Step 7 Enter **rm** and the partition number, and then press **Enter**. In this example, partition number **2** is used.

Information similar to the following is displayed:

```
(parted) rm
Partition number? 2
(parted)
```


Step 8 Run the following command to re-create the partition and press **Enter**:

mkpart *Partition name Start sector End sector*

In this example, run the following command:

mkpart data 209713152s 100%

- Ensure that the entered start sector is the same as the partition had before. In this example, start sector **209713152s** is recorded in **Step 6**. Therefore, enter **209713152s**.
- Ensure that the entered end sector is greater than the partition had before. In this example, the end sector recorded in **Step 6** is **314570751s**, and all the additional space needs to be allocated to **dev/vdb2**. Therefore, enter **100%**.

Information similar to the following is displayed:

```
(parted) mkpart data 209713152s 100%
(parted)
```

NOTE

Data will be lost if the following operations are performed:

- Select a start sector other than the partition had before.
- Select an end sector smaller than the partition had before.

Step 9 Enter **p** and press **Enter** to view the partition information.

Information similar to the following is displayed:

```
(parted) p
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 482344960s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start       End         Size        File system  Name  Flags
 1      2048s       209713151s  209711104s  ext4         test
 2      209713152s  482342911s  272629760s  ext4         data

(parted)
```

Step 10 Perform the following operations based on the file system of the disk:

- For the **ext*** file system
  a. Run the following command to check the correctness of the file system on the partition:

     e2fsck -f *Disk partition*

     In this example, run the following command:

     e2fsck -f /dev/vdb2

     Information similar to the following is displayed:

     ```
     [root@ecs-test-0001 ~]# e2fsck -f /dev/vdb2
     e2fsck 1.42.9 (28-Dec-2013)
     Pass 1: Checking inodes, blocks, and sizes
     Pass 2: Checking directory structure
     Pass 3: Checking directory connectivity
     Pass 4: Checking reference counts
     Pass 5: Checking group summary information
     /dev/vdb2: 11/3276800 files (0.0% non-contiguous), 251790/13107200 blocks
     ```

  b. Run the following command to extend the file system of the partition:

     resize2fs *Disk partition*

In this example, run the following command:

resize2fs /dev/vdb2

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# resize2fs /dev/vdb2
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/vdb2 to 34078720 (4k) blocks.
The filesystem on /dev/vdb2 is now 34078720 blocks long.
```

NOTE

If the error message "open: No such file or directory while opening /dev/vdb1" is returned, an incorrect partition is specified. Run **df -TH** to view the disk partitions.

c. (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

d. Run the following command to mount the partition:

mount *Disk partition Mount point*

In this example, run the following command to mount partition **/dev/vdb2** on **/mnt/test**:

mount /dev/vdb2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

- For the **xfs** file system

a. (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

b. Run the following command to mount the partition:

mount *Disk partition Mount point*

In this example, run the following command to mount partition **/dev/vdb2** on **/mnt/test**:

mount /dev/vdb2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

c. Run the following command to extend the file system of the partition:

sudo xfs_growfs *Disk partition*

In this example, run the following command:

sudo xfs_growfs /dev/vdb2

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# sudo xfs_growfs /dev/vdb2
meta-data=/dev/vdb2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, spinodes=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal               bsize=4096   blocks=6400, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 13107200 to 34078720.
```

Step 11 Run the following command to view 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
/dev/vdb2      ext4      138G   63M  131G   1% /mnt/test
```

----End

Configuring Automatic Mounting at System Start

The **fstab** file controls what disks are automatically mounted at server startup. You can configure the **fstab** file of a server that has data. This operation will not affect the existing data.

The following example uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names (like **/dev/vdb1**) 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 server stop or start. This can even prevent your server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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

5.6.4 Extending Partitions and File Systems for SCSI Disks (Linux)

Scenarios

After a disk is expanded on the management console, the disk size is enlarged, but the additional space cannot be used directly.

In Linux, you must allocate the additional space to an existing partition or a new partition.

This section uses CentOS 7.4 64bit as the sample OS to describe how to extend an MBR partition of a SCSI data disk. The method for allocating the additional space varies with the server OS. This section is used for reference only. For detailed operations and differences, see the corresponding OS documents.

- **Creating a New MBR Partition**
- **Extending an Existing MBR Partition**

NOTICE

Performing the expansion operations with caution. Incorrect operations may lead to data loss or exceptions, so you are advised to back up the disk data using CBR or snapshots before expansion. For details about using CBR, see **Managing EVS Backups**. For details about using snapshots, see the *Elastic Volume Service API Reference*

Prerequisites

- You have expanded the disk capacity and attached the disk to a server on the management console. For details, see **Expanding Capacity for an In-use EVS Disk** or **Expanding Capacity for an Available EVS Disk**.
- You have logged in to the server.
  - For how to log in to an ECS, see the *Elastic Cloud Server User Guide*.
  - For how to log in to a BMS, see the *Bare Metal Server User Guide*.

Creating a New MBR Partition

Originally, data disk **/dev/sda** has 50 GiB and one partition (**/dev/sda1**), and then 50 GiB is added to the disk. The following procedure shows you how to create a new MBR partition **/dev/sda2** with this 50 GiB.

Step 1 Run the following command to view the disk partition information:

fdisk -l

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# 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/sda: 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: 0x915ffe6a
```



```
   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048   104857599    52427776   83  Linux
```

View the **/dev/sda** capacity and check whether the additional space is included.

- If the additional space is not included, refresh the capacity according to **Step 2**.
- If the additional space is included, go to **Step 3**.

Step 2 (Optional) Run the following command to update the capacity of the SCSI data disk:

1. Run the following command to update the disk capacity on the server:

   echo 1 > /sys/class/scsi_device/*%d:%d:%d:%d***/device/rescan &**

   In the command, **%d:%d:%d:%d** indicates a folder in the **/sys/class/scsi_device/** directory and can be obtained using **ll /sys/class/scsi_device/**.

   Information similar to the following is displayed: (**2:0:0:0** indicates the folder to be obtained.)

   ```
   cs-xen-02:/sys/class/scsi_device # ll /sys/class/scsi_device/
   total 0
   lrwxrwxrwx 1 root root 0 Sep 26 11:37 2:0:0:0 -> ../../devices/xen/vscsi-2064/host2/target2:0:0/2:0:0:0/
   scsi_device/2:0:0:0
   ```

   In this example, run the following command:

   echo 1 > /sys/class/scsi_device/2:0:0:0/device/rescan &

2. After the disk capacity is updated, run the following command to view the disk partition information again:

   fdisk -l

   If the additional space is included, go to **Step 3**.

Step 3 Run the following command to enter fdisk:

fdisk *Disk*

In this example, run the following command:

fdisk /dev/sda

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# fdisk /dev/sda
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.


Command (m for help):
```

Step 4 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 (1 primary, 0 extended, 3 free)
   e   extended
Select (default p):
```

There are two types of disk partitions:

- Choosing **p** creates a primary partition.

- Choosing **e** creates an extended partition.

NOTE

If MBR is used, a maximum of 4 primary partitions, or 3 primary partitions plus 1 extended partition can be created. The extended partition must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

Step 5 In this example, a primary partition is created. Therefore, enter **p** and press **Enter** to create a primary partition.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (2-4, default 2):
```

Partition number indicates the serial number of the primary partition. Because partition number 1 has been used, the value ranges from **2** to **4**.

Step 6 Enter the serial number of the primary partition and press **Enter**. Partition number **2** is used in this example. Therefore, enter **2** and press **Enter.**

Information similar to the following is displayed:

```
Partition number (2-4, default 2): 2
First sector (104857600-209715199, default 104857600):
```

First sector indicates the start sector. The value ranges from **104857600** to **209715199**, and the default value is **104857600**.

Step 7 Enter the new partition's start sector and press **Enter**. In this example, the default start sector is used.

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 (104857600-209715199, default 104857600):
Using default value 104857600
Last sector, +sectors or +size{K,M,G} (104857600-209715199, default 209715199):
```

Last sector indicates the end sector. The value ranges from **104857600** to **209715199**, and the default value is **209715199**.

Step 8 Enter the new partition's end sector and press **Enter**. In this example, the default end sector is used.

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} (104857600-209715199, default 209715199):
Using default value 209715199
Partition 2 of type Linux and of size 50 GiB is set

Command (m for help):
```

Step 9 Enter **p** and press **Enter** to view the new partition.

Information similar to the following is displayed:

```
Command (m for help): p
```



```
Disk /dev/sda: 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: 0x915ffe6a

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048   104857599    52427776   83  Linux
/dev/sda2       104857600   209715199    52428800   83  Linux

Command (m for help):
```

Step 10 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.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

NOTE

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

Step 11 Run the following command to synchronize the new partition table to the OS:

partprobe

Step 12 Run the following command to set the file system format for the new partition:

mkfs -t *File system Disk partition*

- Sample command of the ext* file system:

  mkfs -t ext4 /dev/sda2

  Information similar to the following is displayed:

```
[root@ecs-scsi ~]# mkfs -t ext4 /dev/sda2
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
3276800 inodes, 13107200 blocks
655360 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2162163712
400 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

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

- Sample command of the xfs file system:

  mkfs -t xfs /dev/sda2

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# mkfs -t xfs /dev/sda2
meta-data=/dev/sda2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=6400, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

The formatting takes a while, and you need to observe the system running status. Once **done** is displayed in the command output, the formatting is complete.

Step 13 (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

Step 14 Run the following command to mount the new partition:

mount *Disk partition Mount point*

In this example, run the following command to mount the new partition **/dev/sda2** on **/mnt/test**:

mount /dev/sda2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

Step 15 Run the following command to view the mount result:

df -TH

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# df -TH
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/vda1      ext4       43G  2.0G   39G   5% /
devtmpfs       devtmpfs  509M     0  509M   0% /dev
tmpfs          tmpfs     520M     0  520M   0% /dev/shm
tmpfs          tmpfs     520M  7.2M  513M   2% /run
tmpfs          tmpfs     520M     0  520M   0% /sys/fs/cgroup
tmpfs          tmpfs     104M     0  104M   0% /run/user/0
/dev/sda1      ext4       53G   55M   50G   1% /mnt/sdc
/dev/sda2      ext4       53G   55M   50G   1% /mnt/test
```

NOTE

If the server is restarted, the mounting will become invalid. You can modify the **/etc/fstab** file to configure automount at startup. For details, see **Configuring Automatic Mounting at System Start**.

----End

Extending an Existing MBR Partition

NOTICE

If the additional space is allocated to an existing partition, data on the disk will not be cleared but you must use **umount** to unmount the existing partition. In this case, services will be affected.

Originally, SCSI data disk **/dev/sda** has 100 GiB and two partitions (**/dev/sda1** and **/dev/sda2**), and then 50 GiB is added to the disk. The following procedure shows you how to add this 50 GiB to the existing MBR partition **/dev/sda2**.

During an expansion, the additional space is added to the end of the disk. Therefore, if the disk has multiple partitions, the additional space can only be allocated to the partition at the disk end.

Step 1 Run the following command to view the disk partition information:

fdisk -l

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# 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/sda: 161.1 GiB, 161061273600 bytes, 314572800 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: 0x915ffe6a

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048   104857599    52427776   83  Linux
/dev/sda2       104857600   209715199    52428800   83  Linux
```

In the command output, take note of the partition's start and end sectors. In this example, **/dev/sda2**'s start sector is **104857600**, and its end sector is **209715199**.

View the **/dev/sda** capacity and check whether the additional space is included.

- If the additional space is not included, refresh the capacity according to **Step 2**.
- If the additional space is included, take note of the start and end sectors of the target partition and then go to **Step 3**. These values will be used in the subsequent operations.

Step 2 (Optional) Run the following command to update the capacity of the SCSI data disk:

1. Run the following command to update the disk capacity on the server:

   echo 1 > /sys/class/scsi_device/*%d:%d:%d:%d***/device/rescan &**

In the command, **%d:%d:%d:%d** indicates a folder in the **/sys/class/scsi_device/** directory and can be obtained using **ll /sys/class/scsi_device/**.

Information similar to the following is displayed: (**2:0:0:0** indicates the folder to be obtained.)

```
cs-xen-02:/sys/class/scsi_device # ll /sys/class/scsi_device/
total 0
lrwxrwxrwx 1 root root 0 Sep 26 11:37 2:0:0:0 -> ../../devices/xen/vscsi-2064/host2/target2:0:0/2:0:0:0/
scsi_device/2:0:0:0
```

In this example, run the following command:

echo 1 > /sys/class/scsi_device/2:0:0:0/device/rescan &

2. After the disk capacity is updated, run the following command to view the disk partition information again:

   fdisk -l

   If the additional space is included, take note of the start and end sectors of the target partition and then go to **Step 3**. These values will be used in the subsequent operations.

Step 3 Run the following command to unmount the partition:

umount *Disk partition*

In this example, run the following command:

umount /dev/sda2

Step 4 Run the following command to enter fdisk:

fdisk *Disk*

In this example, run the following command:

fdisk /dev/sda

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# fdisk /dev/sda
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.


Command (m for help):
```

Step 5 Run the following command to delete the partition to be extended:

1. Enter **d** and press **Enter** to delete the partition.

   Information similar to the following is displayed:

```
Command (m for help): d
Partition number (1,2, default 2):
```

2. Enter the partition number and press **Enter** to delete the partition. In this example, enter **2**.

   Information similar to the following is displayed:

```
Partition number (1,2, default 2): 2
Partition 2 is deleted

Command (m for help):
```


NOTE

After deleting the partition, re-create the partition according to the following steps, and data on this disk will not be lost.

Step 6 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 (1 primary, 0 extended, 3 free)
   e   extended
Select (default p):
```

There are two types of disk partitions:

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

NOTE

If MBR is used, a maximum of 4 primary partitions, or 3 primary partitions plus 1 extended partition can be created. The extended partition must be divided into logical partitions before use.

Disk partitions created using GPT are not categorized.

Step 7 Ensure that the entered partition type is the same as the partition had before. In this example, a primary partition is used. Therefore, enter **p** and press **Enter** to create a primary partition.

Information similar to the following is displayed:

```
Select (default p): p
Partition number (2-4, default 2):
```

Partition number indicates the serial number of the primary partition.

Step 8 Ensure that entered partition number is the same as the partition had before. In this example, partition number **2** is used. Therefore, enter **2** and press **Enter**.

Information similar to the following is displayed:

```
Partition number (2-4, default 2): 2
First sector (104857600-314572799, default 104857600):
```

In the command output, **First sector** specifies the start sector.

NOTE

Data will be lost if the following operations are performed:

- Select a start sector other than the partition had before.
- Select an end sector smaller than the partition had before.

Step 9 Ensure that the entered start sector is the same as the partition had before. In this example, start sector **104857600** is recorded in **Step 1** or **Step 2**. Therefore, enter **104857600** and press **Enter**.

Information similar to the following is displayed:

```
First sector (104857600-314572799, default 104857600):
Using default value 104857600
Last sector, +sectors or +size{K,M,G} (104857600-314572799, default 314572799):
```

In the command output, **Last sector** specifies the end sector.

Step 10 Ensure that the entered end sector is greater than or equal to the end sector recorded in **Step 1** or **Step 2**. In this example, the recorded end sector is **209715199**, and the default end sector is used. Therefore, enter **314572799** and press **Enter**.

Information similar to the following is displayed:

```
Last sector, +sectors or +size{K,M,G} (104857600-314572799, default 314572799):
Using default value 314572799
Partition 2 of type Linux and of size 100 GiB is set

Command (m for help):
```

The partition is created.

Step 11 Enter **p** and press **Enter** to print the partition details.

Information similar to the following is displayed:

```
Command (m for help): p

Disk /dev/sda: 161.1 GiB, 161061273600 bytes, 314572800 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: 0x915ffe6a

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048   104857599    52427776   83  Linux
/dev/sda2       104857600   314572799   104857600  83  Linux

Command (m for help):
```

Step 12 Enter **w** and press **Enter** to write the changes to the partition table.

Information similar to the following is displayed: (The partition is successfully created.)

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

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

NOTE

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

Step 13 Run the following command to synchronize the new partition table to the OS:

partprobe

Step 14 Perform the following operations based on the file system of the disk:

- For the **ext***file system
  a. Run the following command to check the correctness of the file system on the partition:

     e2fsck -f *Disk partition*

     In this example, run the following command:

     e2fsck -f /dev/sda2

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# e2fsck -f /dev/sda2
e2fsck 1.42.9 (28-Dec-2013)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/sda2: 11/3276800 files (0.0% non-contiguous), 251790/13107200 blocks
```

b. Run the following command to extend the file system of the partition:

resize2fs *Disk partition*

In this example, run the following command:

resize2fs /dev/sda2

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# resize2fs /dev/sda2
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/sda2 to 26214400 (4k) blocks.
The filesystem on /dev/sda2 is now 26214400 blocks long.
```

c. (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

d. Run the following command to mount the partition:

mount *Disk partition Mount point*

In this example, run the following command to mount partition **/dev/sda2** on **/mnt/test**:

mount /dev/sda2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

- For the **xfs** file system

a. (Optional) Run the following command to create a mount point:

Perform this step if you want to mount the partition on a new mount point.

mkdir *Mount point*

In this example, run the following command to create the **/mnt/test** mount point:

mkdir /mnt/test

b. Run the following command to mount the partition:

mount *Disk partition Mount point*

In this example, run the following command to mount partition **/dev/sda2** on **/mnt/test**:

mount /dev/sda2 /mnt/test

NOTE

If the new partition is mounted on a directory that is not empty, the subdirectories and files in the directory will be hidden. Therefore, you are advised to mount the new partition on an empty directory or a new directory. If the new partition must be mounted on a directory that is not empty, move the subdirectories and files in this directory to another directory temporarily. After the partition is successfully mounted, move the subdirectories and files back.

c. Run the following command to extend the file system of the partition:

sudo xfs_growfs *Disk partition*

In this example, run the following command:

sudo xfs_growfs /dev/sda2

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# sudo xfs_growfs /dev/sda2
meta-data=/dev/sda2              isize=512    agcount=4, agsize=3276800 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, spinodes=0
data     =                       bsize=4096   blocks=13107200, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal               bsize=4096   blocks=6400, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 13107200 to 26214400df .
```

Step 15 Run the following command to view the mount result:

df -TH

Information similar to the following is displayed:

```
[root@ecs-scsi ~]# df -TH
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/vda1      ext4       43G  2.0G   39G   5% /
devtmpfs       devtmpfs  509M     0  509M   0% /dev
tmpfs          tmpfs     520M     0  520M   0% /dev/shm
tmpfs          tmpfs     520M  7.2M  513M   2% /run
tmpfs          tmpfs     520M     0  520M   0% /sys/fs/cgroup
tmpfs          tmpfs     104M     0  104M   0% /run/user/0
/dev/sda1      ext4       53G   55M   50G   1% /mnt/sdc
/dev/sda2      ext4      106G   63M  101G   1% /mnt/test
```

----End

Configuring Automatic Mounting at System Start

The **fstab** file controls what disks are automatically mounted at server startup. You can configure the **fstab** file of a server that has data. This operation will not affect the existing data.

The following example uses UUIDs to identify disks in the **fstab** file. You are advised not to use device names (like **/dev/vdb1**) 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 server stop or start. This can even prevent your server from booting up.

NOTE

UUIDs are the unique character strings for identifying partitions in Linux.

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
```

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 Detaching an EVS Disk

6.1 Detaching a System Disk

Scenarios

If the file system on your system disk is damaged and your server cannot be started, you can detach the system disk and attach it to another server as a data disk. After the file system is fixed, you can re-attach the disk to the original server as the system disk.

If you no longer need a system disk or want to replace it with a new one, you can detach the system disk.

A system disk can only be detached offline, which means that its server must be in the **Stopped** state before the system disk is detached. To detach a system disk from a running server, you must first stop the server and then detach the disk.

NOTE

- For an attached system disk, the disk function is displayed as **System disk**, and the disk status is displayed as **In-use** in the disk list. After a system disk is detached from the server, the disk function changes to **Bootable disk**, and the status changes to **Available**.
- 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.

Constraints

- A system disk can only be detached offline, which means that its server must be in the **Stopped** state before you can detach it. To detach a system disk from a running server, you must first stop the server and then detach the disk.
- After a system disk is detached, some operations cannot be performed on the original server and the system disk. The restricted operations are as follows:
  – Server: starting the server, remote login, resetting the password, changing server billing mode, changing server specifications, changing the OS, reinstalling the OS, creating images, creating backups, adding disks, changing the security group, and changing the VPC

Procedure

Step 1 Log in to the management console.

Step 2 Under **Compute**, click **Elastic Cloud Server**.

The **Elastic Cloud Server** page is displayed.

Step 3 In the server list, locate the row that contains the server whose system disk is to be detached, click **More** in the **Operation** column, and choose **Stop**.

When the server status changes to **Stopped**, the server has been stopped.

Step 4 Click the name of this server.

The server details page is displayed.

Step 5 Click the **Disks** tab to view the system disk attached to the server.

Step 6 Locate the row that contains the system disk and click **Detach**.

The **Detach Disk** dialog box is displayed.

Step 7 Click **OK** to detach the disk.

After the operation had succeeded, the detached system disk is no longer displayed under the **Disks** tab.

Step 8 (Optional) Re-attach the bootable disk to a server. You can use it as a system disk or data disk depending on the disk function you select.

- To re-attach and use it as a system disk, see **Attaching an Existing System Disk**.
- To re-attach and use it as a data disk, see **Attaching an Existing Non-Shared Disk**.

----End

Related Operations

For more detachment FAQs, see **Detachment**.

6.2 Detaching a Data Disk

Scenarios

If you want to use a data disk on another server in the same region and AZ, you can detach the data disk and then attach it to that server.

If a data disk is no longer required, you can detach it and then delete it.

Data disks can be detached online or offline, meaning that the server using the to-be-detached data disk can either be in the **Running** or **Stopped** state.

- ECS

  Detach a disk from a running server. For details, see **Storage** > **Detaching an EVS Disk from a Running ECS** in the *Elastic Cloud Server User Guide*.

- BMS

  SCSI disks can be attached to BMSs and used as data disks. You can detach a data disk either from a running or stopped BMS.

NOTE

For an attached data disk, the disk function is displayed as **Data disk**, and the disk status is displayed as **In-use** in the disk list. After the data disk has been detached from the server, the disk function remains unchanged, the disk status changes to **Available** for a non-shared data disk, and the disk status changes to **Available** for a shared data disk after it is detached from all its servers.

Precautions

Data may be lost after you detach an encrypted disk. For more information, see **If I Detach a Disk, Will I Lose the Data on My Disk?**.

Prerequisites

- Before detaching an EVS disk from a running Windows ECS, ensure that no programs are reading data from or writing data to the disk. Otherwise, data will be lost.
- Before detaching an EVS disk from a running Linux ECS, you must log in to the ECS and run the **umount** command to cancel the association between the disk and the file system. In addition, ensure that no programs are reading data from or writing data to the disk. Otherwise, detaching the disk will fail.

Detaching a Non-shared Disk

Step 1 Log in to the management console.

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

The disk list page is displayed.

Step 3 Choose a way to detach the disk by determining whether you want to check server information first.

- If yes, perform the following procedure:
  a. In the disk list, click the name of the to-be-detached disk.

     The disk details page is displayed.
  b. Click the **Attachments** tab to view the server where the target disk has been attached.
  c. Click to select the server and click **Detach Disk**.

     The **Detach Disk** dialog box is displayed.
  d. Click **OK** to detach the disk.
- If no, perform the following procedure:
  a. In the disk list, locate the row that contains the target disk and click **Detach** in the **Operation** column.

     The **Detach Disk** dialog box is displayed.
  b. Click **OK** to detach the disk.

The disk list is displayed. The disk status is **Detaching**, indicating that the disk is being detached from the server.

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

----End

Detaching a Shared Disk

Step 1 Log in to the management console.

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

The disk list page is displayed.

Step 3 Choose a way to detach the disk by determining whether you want to check server information first.

- If yes, perform the following procedure:
  a. In the disk list, click the name of the to-be-detached disk.
     The disk details page is displayed.
  b. Click the **Attachments** tab to view the servers where the target disk has been attached.
  c. Click to select the server and click **Detach Disk**.
     Shared EVS disks support batch detachment so that you can select multiple servers at a time.
     The **Detach Disk** dialog box is displayed.
  d. Click **OK** to detach the disk.
- If no, perform the following procedure:
  a. In the disk list, locate the row that contains the target disk and click **Detach** in the **Operation** column.
     The **Detach Disk** dialog box is displayed.
  b. Click to select the server.
     Shared EVS disks support batch detachment so that you can select multiple servers at a time.
  c. Click **OK** to detach the disk.

The disk list page is displayed. The disk status is **Detaching**, indicating that the disk is being detached from the server.

If the shared EVS disk has been attached to multiple servers and needs to be detached from only some of its servers, the disk status will go back to **In-use** after the disk has been detached from the target servers. The disk status changes to **Available** only when it has been detached from all the servers.

----End

7 Attaching an Existing Disk

7.1 Attaching an Existing System Disk

Scenarios

This section describes how to attach an existing system disk.

System disks can only be attached offline, which means that the server must be in the **Stopped** state.

You can view the disk function in the disk list. A disk can be attached to a server and used as the system disk only when its function is **Bootable disk** and its status is **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.

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 shared disk can be attached only when the servers' statuses are **Running** or **Stopped**.
- A frozen disk cannot be attached.

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

Step 4 Select a server to attach the disk. Ensure that the disk and server are in the same AZ and that the server is in the **Stopped** state. After you select a server, **System disk** is automatically entered as the disk function for the server.

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*.

Step 5 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 function of the disk changes from **Bootable disk** to **System disk** and the disk status changes to **In-use**, the disk is successfully attached.

----End

Related Operations

For more attachment FAQs, see **Attachment**.

7.2 Attaching an Existing Non-Shared Disk

Scenarios

This section describes how to attach an existing non-shared disk to a server and use it as a data disk. A non-shared disk can be attached to one server only.

You can view the disk information in the disk list. A disk can be attached to a server and used as a data disk when all of the following conditions are met:

- Disk Sharing: Disabled
- Function: Bootable disk or Data disk
- Status: 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.

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 shared disk can be attached only when the servers' statuses are **Running** or **Stopped**.
- A frozen disk cannot be attached.

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

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

Step 5 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 6 (Optional) Mount the existing disk partition to a mount point if you have attached the disk to a Linux server. The mount command is as follows:

```
mount Disk partition Mount point
```

----End

Related Operations

For more attachment FAQs, see **Attachment**.

7.3 Attaching an Existing Shared Disk

Scenarios

This section describes how to attach an existing shared disk to a server and use it as a data disk.

You can view the disk information in the disk list. A shared disk can be attached to servers and used as a data disk when all of the following conditions are met:

- Disk Sharing: Enabled
- Function: Data disk
- Status: In-use or Available

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.

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 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.
- A shared disk can be attached to servers only when the disk status is **Available** or **In-use**.

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.

Step 5 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.

Step 6 (Optional) Mount the existing disk partition to a mount point if you have attached the disk to a Linux server. The mount command is as follows:

```
mount Disk partition Mount point
```

----End

Related Operations

For more attachment FAQs, see **Attachment**.

8 Deleting EVS Disks

Scenarios

If an EVS disk is no longer used, you can release the virtual resources by deleting it.

After a disk is deleted, you are no longer charged for it. EVS immediately destroys the metadata so that data can no longer be accessed. In addition, the physical storage space of the disk is reclaimed and cleared before being re-assigned. For any new disk created based on the re-assigned physical space, before data is written to the disk, EVS returns zero for all the read and write requests to the disk.

- Pay-per-use disk:

  It can only be deleted when the following conditions are met:

  - The disk status is **Available**, **Error**, **Expansion failed**, **Restoration failed**, or **Rollback failed**.
  - The disk is not added to any replication pair in the Storage Disaster Recovery Service (SDRS). For any disk already added to a replication pair, you need to first delete the replication pair by referring to section "Deleting a Replication Pair" in the *Storage Disaster Recovery Service User Guide* and then delete the disk.
  - The disk is not locked by any service.
  - The shared disk has been detached from all its servers.

NOTICE

When you delete a disk, all the disk data will be deleted.

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 In the disk list, locate the row that contains the target disk, click **More** in the **Operation** column, and choose **Delete**.

Step 4 (Optional) If multiple disks are to be deleted, select in front of each disk and click **Delete** in the upper area of the list.

Step 5 In the displayed dialog box, confirm the information and click **OK**.

----End

Related Operations

For more deletion FAQs, see **Deletion**.

9 Viewing EVS Disk Details

Scenarios

This section describes how to view disk details, including the disk status and specifications. You can view disk details:

- **From the EVS Console**
- **From the Cloud Server Console**

See **EVS Disk Status** to learn more about disk statuses.

From 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 In the disk list, view disk information including the disk status, type, size, function, and device type.

You can filter disks by project, status, disk name, or tag.

Step 4 In the disk list, locate the desired disk and click the disk name.

The disk details page is displayed for you to view the disk details.

Step 5 (Optional) Export disk information.

Click  in the upper right corner of the list to export disk information.

----End

From the Cloud Server Console

Step 1 Log in to the management console.

Step 2 Under **Compute**, click **Elastic Cloud Server**.

The **Elastic Cloud Server** page is displayed.

Step 3 In the server list, locate the desired server by server name and click the name.

The server details page is displayed.

Step 4 On the **Disks** tab, click in front of the row containing the target disk. In the unfolded area, click the disk ID.

The disk details page is displayed for you to view the disk details.

----End

10 Managing Encrypted EVS Disks

Encryption Scenarios

- **System disk encryption**

  System disks are created along with servers and cannot be created separately. So whether a system disk is encrypted or not depends on the image selected during the server creation. See the following table for details.

Table 10-1 Encryption relationship between images and system disks

Creating Server Using Encrypted Image	Whether System Disk Will Be Encrypted	Description
Yes	Yes	For details, see **Managing Private Images** > **Encrypting Images** in the *Image Management Service User Guide*.
No	No	-

- **Data disk encryption**

  Data disks can be created along with servers or separately. Whether data disks are encrypted depends on their data sources. See the following table for details.

Table 10-2 Encryption relationship between backups, images, and data disks

Created On	Method of Creation	Whether Data Disk Will Be Encrypted	Description
The ECS console	Created together with the server	Yes/No	When a data disk is created together with a server, you can choose to encrypt the disk or not. For details, see **Getting Started** > **Creating an ECS** > **Step 1: Configure Basic Settings** in the *Elastic Cloud Server User Guide*.
The EVS console	No data source selected	Yes/No	When an empty disk is created, you can choose whether to encrypt the disk or not. The encryption attribute of the disk cannot be changed after the disk has been created.
	Creating from a backup	Yes/No	● When a disk is created from a backup, you can choose whether to encrypt the disk or not. The encryption attributes of the disk and backup do not need to be the same. ● When you create a backup for a system or data disk, the encryption attribute of the backup will be the same as that of the disk.
	Creating from an image (The image's source disk is encrypted.)	Yes	-
	Creating from an image (The image's source disk is not encrypted.)	No	-

Constraints

Table 10-3 Constraints on disk encryption

Item	Description
Types of disks supporting encryption	All disk types
Constraints on encrypted disks	The encryption attribute of a disk cannot be changed after the disk is created, meaning that: ● An encrypted disk cannot be changed to a non-encrypted disk. ● A non-encrypted disk cannot be changed to an encrypted disk.
Constraints on user permissions	When a user uses the encryption function, the condition varies depending on whether the user is the first one ever in the current region or project to use this function. ● If the user is the first user, the user needs to follow the prompt to create an agency, which grants KMS Administrator permissions to EVS. Then the user can create and obtain keys to encrypt and decrypt disks. **NOTE** The first user must have the KMS Administrator permissions to create the agency. If the user does not have the KMS Administrator permissions, contact the account administrator to grant the permissions first. ● If the user is not the first user, the user can use encryption directly.
Constraints on encrypted images	● Encrypted images cannot be replicated across regions. ● Encrypted images cannot be changed to non-encrypted images. ● Encrypted images cannot be exported.

Creating an Encrypted EVS Disk

Before you use the encryption function, KMS access rights need to be granted to EVS. If you have the Security Administrator permissions, grant the KMS access rights to EVS directly. If you do not have this permission, contact a user with the security administrator permissions to grant KMS access rights to EVS and then select the encryption option to create an encrypted disk.

For details about how to create an encrypted disk, see **Create an EVS Disk**.

Detaching an Encrypted EVS Disk

Before you detach a disk encrypted by a CMK, check whether the CMK is disabled or scheduled for deletion.

- If the CMK is available, the disk can be detached and re-attached, and data on the disk will not be lost.
- If the CMK is unavailable, the disk can still be used, but there is no guarantee for how long it will be usable. If the disk is detached, it will be impossible to re-attach it later. In this case, do not detach the disk without a working CMK.

The restoration method varies depending on the CMK status. For details, see **EVS Encryption**.

For details about how to detach an encrypted disk, see **Detaching a Data Disk**.

11 Managing Shared EVS Disks

How to Use Shared VBD and SCSI Disks

You can create shared VBD disks or shared SCSI disks. It is recommended that you attach a shared disk to the ECSs in the same ECS group to improve service reliability.

- Shared VBD disks: The device type of a newly created shared disk is VBD by default. Such disks can be used as virtual block storage devices, but do not support SCSI reservations. If SCSI reservations are required for your applications, create shared SCSI EVS disks.
- Shared SCSI disks: Such disks support SCSI reservations.

NOTICE

- To improve data security, you are advised to use SCSI reservations together with the anti-affinity policy of an ECS group. That said, ensure that shared SCSI disks are only attached to ECSs in the same anti-affinity ECS group.
- If an ECS does not belong to any anti-affinity ECS group, you are advised not to attach shared SCSI disks to this ECS. Otherwise, SCSI reservations may not work properly, which may put your data at risk.

Concepts of the anti-affinity ECS group and SCSI reservations:

- The anti-affinity policy of an ECS group allows ECSs to be created on different physical servers to improve service reliability.

  For details about ECS groups, see **Managing ECS Groups** in the *Elastic Cloud Server User Guide*.
- The SCSI reservation mechanism uses a SCSI reservation command to perform SCSI reservation operations. If an ECS sends such a command to an EVS disk, the disk is displayed as locked to other ECSs, preventing the data damage that may be caused by simultaneous read/write operations to the disk from multiple ECSs.
- ECS groups and SCSI reservations have the following relationship: A SCSI reservation on a single EVS disk cannot differentiate multiple ECSs on the same physical host. For that reason, if multiple ECSs that use the same shared EVS disk are running on the same physical host, SCSI reservations

will not work properly. So you are advised to use SCSI reservations only on ECSs that are in the same ECS group, thus having a working anti-affinity policy.

Constraints on Shared Disks

- A shared disk can be attached to a maximum of 16 servers.
- The sharing attribute of a disk cannot be changed after the disk is created.
- Shared disks can only be used as data disks. The sharing function is not supported for system disks.
- A shared file system or cluster management system must be set up before you can properly use a shared disk. If you simply attach a shared disk to multiple servers, the sharing function will not work and data may be overwritten.
- When a shared disk is attached to multiple servers, the total performance of the disk on all servers cannot exceed the maximum allowed on a single disk.

Attaching a Shared EVS Disk

A non-shared EVS disk can only be attached to one server, whereas a shared EVS disk can be attached to up to 16 servers.

For details, see **Attaching a Shared Disk**.

Deleting a Shared EVS Disk

Because a shared EVS disk can be attached to multiple servers, ensure that the shared EVS disk is detached from all the servers before deletion.

For details, see **Deleting EVS Disks**.

Expanding a Shared EVS Disk

Shared EVS disks must be expanded when they are in the **Available** state. For details, see **Expanding Capacity for an In-use EVS Disk**.

Related Operations

For more disk sharing FAQs, see **Sharing**.

12 Managing EVS Backups

Scenarios

EVS backups are created using the VBS service. For details, see **Creating a VBS Backup** in the *Volume Backup Service User Guide*.

You can configure a backup policy for disks. With backup policies configured, data on EVS disks can be periodically backed up to improve data security.

Constraints

- Backups can be created only when the disks are in the **Available** or **In-use** state.

Creating a Disk Backup Vault and Applying a Backup Policy

Step 1 Log in to the CBR console.

1. Log in to the management console.
2. Choose **Storage** > **Cloud Backup and Recovery** > **Cloud Disk Backups**.

Step 2 In the upper right corner, click **Create Disk Backup Vault**.

Step 3 (Optional) In the disk list, select the disks you want to back up. After disks are selected, they are added to the list of selected disks.

NOTE

- Only **Available** and **In-use** disks can be selected.
- You can also associate disks with the vault you are creating later if you skip this step.

Step 4 Specify a vault capacity ranging from the total sizes of disks to 10,485,760 GiB.

Step 5 Configure auto backup.

- If you select **Configure**, you must then select an existing backup policy or create a new one. After the vault is created, the system applies this backup policy to the vault, and all disks associated with this vault will be automatically backed up based on this policy.
- If you select **Skip**, disks associated with this vault are not automatically backed up.

Step 6 Configure auto capacity expansion.

- If you select **Configure**, the vault capacity will be automatically expanded to 1.25 times the size of the original capacity when the capacity is about to be used up.
- If you select **Skip**, the vault capacity will not be expanded automatically.

Step 7 If you have subscribed to Enterprise Project, add the vault to an existing enterprise project.

An enterprise project facilitates project-level management and grouping of cloud resources and users. The default project is **default**.

Step 8 (Optional) Add tags to the vault.

A tag consists of a key-value pair. Tags are used to identify, classify, and search for vaults. Vault tags are used to filter and manage vaults only. You can add up to 10 tags for a vault.

Table 12-1 describes the tag parameters.

Table 12-1 Tag parameters

Parameter	Description	Example Value
Key	A tag key of a vault must be unique. You can customize the key or select the key of an existing tag created in TMS. A tag key: ● Can contain 1 to 36 Unicode characters. ● Cannot be left blank, cannot start or end with spaces, or contain non-printable ASCII (0-31) characters or the following special characters: =*<>\,|/	Key_0001
Value	A tag value can be repetitive or left blank. A tag value: ● Can contain 0 to 43 Unicode characters. ● Can be an empty string, cannot start or end with spaces, or contain non-printable ASCII (0-31) characters or the following special characters: =*<>\,|/	Value_0001

Step 9 Specify a name for the vault.

The name can contain 1 to 64 characters including digits, letters, underscores (_), and hyphens (-), for example, **vault-612c**.

NOTE

You can use the default name, which is in the format of **vault**_*xxxx*.

Step 10 Click **Next**.

Step 11 Complete the creation as prompted.

Step 12 Go back to the disk backup page. The vault you created is displayed in the list.

You can associate disks to the new vault or create backups for the disks. For details, see .

----End

13 Managing a Tag

13.1 Tag Overview

Tags identify EVS resources for purposes of easy categorization and quick search.

Table 13-1 Tag overview

Operation	Scenario
Adding a Tag	Add tags for existing disks or during disk creations.
Modifying a Tag	Change tag values for existing disks. Tag keys of existing disks cannot be changed.
Deleting a Tag	Delete tags that are no longer needed for existing disks.
Searching for Disks by Tag	After tags are added, search for disks by tags.

13.2 Adding a Tag

Scenarios

You can add a tag for an existing EVS disk. You can also add tags during the disk creation. For details, see **Create an EVS Disk**.

- A tag is composed of a key-value pair.
  - A tag key can contain a maximum of 36 characters, including letters, digits, and underscores (_).
  - A tag value can contain a maximum of 43 characters, including letters, digits, underscores (_), periods (.), and hyphens (-).
- A maximum of 10 tags can be added for an EVS disk.

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 In the disk list, locate the desired disk and click the disk name.

The disk details page is displayed.

Step 4 Click the **Tags** tab.

Step 5 Click **Add Tag**.

The **Add Tag** page is displayed.

Step 6 Enter a key and a value for a tag and click **OK**.

- **Key**: This parameter is mandatory.
- **Value**: This parameter is optional.

The **Tags** tab is displayed, and you can view the newly added tag.

----End

13.3 Modifying a Tag

Scenarios

You can change the value of a tag for an existing disk, but cannot change the key of a tag.

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 In the disk list, locate the desired disk and click the disk name.

The disk details page is displayed.

Step 4 Click the **Tags** tab.

Step 5 Locate the target tag and click **Edit** in the **Operation** column.

The **Edit Tag** page is displayed.

Step 6 Change the value of the tag and click **OK**.

Return to the tag list. If the tag value is changed, the modification is complete.

----End

13.4 Deleting a Tag

Scenarios

If an existing tag is no longer needed, you can delete it.

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 In the disk list, locate the desired disk and click the disk name.

The disk details page is displayed.

Step 4 Click the **Tags** tab.

Step 5 Locate the target tag and click **Delete** in the **Operation** column.

The **Delete Tag** page is displayed.

Step 6 Confirm the information and click **OK**.

The **Tags** tab is displayed, and the deletion is complete.

----End

13.5 Searching for Disks by Tag

Scenarios

Tags can be used to categorize EVS disks, and users can quickly search for their desired EVS disks by tags. This section is used to guide users to search for EVS disk by existing tags.

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 In the upper area of the disk list, click **Search by Tag**.

The **Search by Tag** page is displayed.

Step 4 Enter or select an existing tag in the text box under **Search by Tag**.

Step 5 (Optional) If disks containing multiple tags need to be queried, click  to add tags.

You can add a maximum of 10 tags to search for disks. If you add more than one tag, only the disks containing all specified tags will be returned.

For the added tags, you can delete them individually or click **Reset** to clear all of them.

Step 6 After the tags are added, click **Search**.

Disks owning the added tags are displayed in the list, and the search is complete.

----End

14 Changing EVS Disk Name

Scenarios

Disk names are used to identify disks. After a disk is created, you can perform operations in this section to change the disk name if needed.

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 Change the disk name in either of the following ways:

- Perform the following steps to change the disk name in the disk list:
  a. In the disk list, locate the target disk in the **Disk Name** column and click to the right of the disk name.
     The **Edit Disk Name** dialog box is displayed.
  b. Enter a new name.
  c. Click **OK**.
     After the change is successful, the new disk name is displayed in the disk list.
- Perform the following steps to change the disk name on the disk details page:
  a. In the disk list, locate the desired disk and click the disk name.
     The disk details page is displayed.
  b. Click next to the disk name.
  c. Enter a new name.
  d. Click .
     After the change is successful, the new disk name is displayed on the disk details page.

----End

15 Viewing EVS Monitoring Data

Description

This section describes monitored metrics reported by EVS to Cloud Eye as well as their namespaces and dimensions. You can use the console or APIs provided by Cloud Eye to query the metrics of the monitored objects and alarms generated for EVS.

Namespace

SYS.EVS

Metrics

Table 15-1 EVS metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period
disk_device_read_bytes_rate	Disk Read Bandwidth	Number of bytes read from the monitored disk per second Unit: Bytes/s	≥ 0 bytes/s	EVS disk	5 minutes in average
disk_device_write_bytes_rate	Disk Write Bandwidth	Number of bytes written to the monitored disk per second Unit: Bytes/s	≥ 0 bytes/s	EVS disk	5 minutes in average
disk_device_read_requests_rate	Disk Read IOPS	Number of read requests sent to the monitored disk per second Unit: Requests/s	≥ 0 Requests/s	EVS disk	5 minutes in average

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period
disk_device_write_requests_rate	Disk Write IOPS	Number of write requests sent to the monitored disk per second Unit: Requests/s	≥ 0 Requests/s	EVS disk	5 minutes in average
disk_device_queue_length	Average Queue Length	Average number of read or write requests waiting for processing in the monitoring period for the monitored disk Unit: Count	≥ 0 Counts	EVS disk	5 minutes in average
disk_device_io_util	Disk I/O Utilization	Percentage of time spent during which read and write requests were sent to the monitored disk in the monitoring period Unit: Percent	0-100%	EVS disk	5 minutes in average
disk_device_write_bytes_per_operation	Avg Disk Bytes Per Write	Average number of bytes transmitted per I/O write for the monitored disk in the monitoring period Unit: Kbyte/operation	≥ 0 KiB/op	EVS disk	5 minutes in average
disk_device_read_bytes_per_operation	Avg Disk Bytes Per Read	Average number of bytes transmitted per I/O read for the monitored disk in the monitoring period Unit: Kbyte/operation	≥ 0 KiB/op	EVS disk	5 minutes in average
disk_device_write_await	Disk Write Await	Average await time per I/O write for the monitored disk in the monitoring period Unit: ms/operation	≥ 0 ms/operation	EVS disk	5 minutes in average

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period
disk_device_read_await	Disk Read Await	Average await time per I/O read for the monitored disk in the monitoring period Unit: ms/operation	≥ 0 ms/operation	EVS disk	5 minutes in average
disk_device_io_svctm	Disk I/O Service Time	Average service time per I/O read or write for the monitored disk in the monitoring period Unit: ms/operation	≥ 0 ms/operation	EVS disk	5 minutes in average
disk_device_io_iops_qos_num	IOPS Upper Limit Reached (Count)	Number of times that the IOPS of the monitored disk has reached the upper limit Unit: Count	≥ 0 Counts	EVS disk	5 minutes in average
disk_device_io_iobw_qos_num	Bandwidth Upper Limit Reached (Count)	Number of times that the bandwidth of the monitored disk has reached the upper limit Unit: Count	≥ 0 Counts	EVS disk	5 minutes in average

Dimension

Key	Value
disk_name	Server ID-drive letter, for example, 6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d-vda (vda is the drive letter)

Viewing Monitoring Data

Step 1 Log in to the management console.

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

The disk list page is displayed.

Step 3 In the EVS disk list, click the name of the disk you want to view the monitoring data.

The disk details page is displayed.

Step 4 On the **Attachments** tab, locate the row that contains the server and click **View Monitoring Data** in the **Operation** column.

The monitoring graphs page is displayed.

Step 5 View the disk monitoring data by metric or monitored duration.

For more information about Cloud Eye, see the *Cloud Eye User Guide*.

----End

16 Viewing EVS Monitoring Data (Agent Installed and Simplified Monitoring Metrics Used)

Description

This section describes the EVS-related metrics included in the OS metrics supported by ECS. The agent of the latest version is used with simplified monitoring metrics.

After installing the agent on an ECS, you can view its EVS-related metrics included in the OS monitoring metrics.

For instructions about how to install and configure the agent, see section "Agent Installation and Configuration" in the *Cloud Eye User Guide*.

Monitoring Metrics

Table 16-1 EVS-related metrics

Metric	Name	Description	Value Range	Monitored Object	Monitoring Period
mountPointPrefix_disk_free	(Agent) Available Disk Space	Available disk space on the monitored object Unit: GiB • Linux: Run the **df -h** command to check the value in the **Avail** column. The path of the device name prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), periods (.), and swung dashes (~). • Windows: Obtain the metric value using the WMI API GetDiskFreeSpaceExW. The path of the device name prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), periods (.), and swung dashes (~).	≥0 GiB	ECS	5 minutes in average

Metric	Name	Description	Value Range	Monitored Object	Monitoring Period
mountPointPrefix_disk_usedPercent	(Agent) Disk Usage	Percentage of total disk space that is used, which is calculated as follows: **Disk Usage** = **Used Disk Space**/ **Disk Storage Capacity**. Unit: Percent ● Linux: Obtain the metric value using the following formula: **Disk Usage** = **Used Disk Space**/**Disk Storage Capacity**. The path of the device name prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), periods (.), and swung dashes (~). ● Windows: Obtain the metric value using the WMI API GetDiskFreeSpaceExW. The path of the device name prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), periods (.), and swung dashes (~).	0-100%	ECS	5 minutes in average

Metric	Name	Description	Value Range	Monitored Object	Monitoring Period
mountPointPrefix_disk_ioUtils and volumePrefix_disk_ioUtils	(Agent) Disk I/O Usage	Percentage of the time that the disk has had I/O requests queued to the total disk operation time Unit: Percent ● Linux: The disk I/O usage is calculated based on the data changes in the thirteenth column of the corresponding device in file **/proc/diskstats** in a collection period. The path of the device name prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), periods (.), and swung dashes (~). ● Windows does not support this metric.	0-100%	ECS	5 minutes in average

Dimensions

Key	Value
instance_id	Specifies the ECS ID.

17 Auditing

Scenarios

EVS supports the recording of EVS operations through CTS. You can query EVS traces and use them for historical operation audits and backtracks.

Prerequisites

CTS has been enabled.

Key EVS Operations Recorded by CTS

Table 17-1 EVS operations that can be recorded by CTS

Operation	Resource	Trace
Create disk	evs	createVolume
Update disk	evs	updateVolume
Expand disk capacity	evs	extendVolume
Delete disk	evs	deleteVolume
Batch create disks	evs	batchCreateVolume
Modify disk specifications	evs	retypeVolume
Prepare for server stop	evs	prepareForStopServer
Batch delete disks	evs	bulkDeleteVolume
Clear EVS resources	evs	cleanupEvsResources
Clear a single EVS resource	evs	cleanupSingleEvsResource
Update disk status	evs	updateVolumesBusinessSta-tus

Operation	Resource	Trace
Unfreeze disk	evs	unfreezeVolume
Freeze disk	evs	freezeVolume
Release disk	evs	releaseVolume

Viewing Traces

To view audit logs, see **Querying Real-Time Traces** in the *Cloud Trace Service User Guide.*

18 Managing Quotas

18.1 Querying EVS Resource Quotas

Scenarios

Quotas are enforced for service resources on the platform to prevent unforeseen spikes in resource usage. Quotas can limit the number or amount of resources available to users, such as the number of EVS disks, the capacity of EVS disks, and the number of EVS snapshots.

Users can perform the following operations to view the resource quota details.

Procedure

Step 1 Log in to the management console.

Step 2 In the upper right corner of the page, click  .

The **Service Quota** page is displayed.

Step 3 View the used and total quota of each type of resources on the displayed page.

If a quota cannot meet service requirements, apply for a higher quota.

----End

18.2 Increasing EVS Resource Quotas

Scenarios

If any resource quota no longer meets your service requirements, you can apply for a higher quota.

How Do I Apply for a Higher Quota?

The system does not support online quota adjustment.

If you need to adjust a quota, contact the operations administrator.

19 FAQ

19.1 Summary

General

- How Do I Start Using a Newly Created Disk?
- Can EVS Disks Be Used Directly for Storage?
- Can EVS Disks Be Used Alone?
- How Can I View My Disk Details?
- Can I Change the AZ of My Disk?
- Can I Change the Disk Type, Device Type, or Sharing Attribute of My Disk?
- What Should I Do If an Error Occurs on My EVS Disk?
- Why Do Some of My EVS Disks Not Have WWN Information?
- How Can I Migrate Data from an EVS Disk?
- What Are the Differences Between System Disks and Data Disks?
- How Can I Upload Files to My EVS Disk?

Capacity Expansion

- Can I Reduce or Temporarily Expand the Disk Capacity?
- What Are the Differences Between Expanding Capacity by Expanding an EVS Disk and Creating a New EVS Disk?
- Will My Disk Data Be Lost After I Expand the Disk Capacity?
- Can I Use Backups Created Before Capacity Expansion to Restore Data on Expanded Disks?
- Do I Need to Restart the Server After Expanding the Disk Capacity?
- Do I Need to Detach an EVS Disk Before Expanding Its Capacity?
- What Should I Do If My Disk Capacity Exceeds 2 TiB After Expansion?
- How Can I Allocate Newly Added Space to a New Partition?
- How Can I Allocate Newly Added Space to an Existing Partition?

Attachment

Detachment

Deletion

Capacity

Performance

Sharing

- How Many Servers Can I Attach a Shared Disk to?
- How Can I Attach a Shared Disk to Multiple Servers?
- Can I Attach a Shared Disk to Servers Running Different OSs?

Backup

- Do I Need to Stop the Server Before Performing a Backup?
- Can I Back Up and Restore My EVS Disk to a Different Region?

19.2 General

19.2.1 How Do I Start Using a Newly Created Disk?

A newly purchased disk must be attached to a server and then initialized in the server OS before you can use it.

For more information, see **Process Overview**.

19.2.2 Can EVS Disks Be Used Directly for Storage?

No.

EVS disks must be attached to cloud servers before use. You cannot use EVS disks alone to store data.

19.2.3 Can EVS Disks Be Used Alone?

No.

EVS disks must be attached to servers before you can use them.

19.2.4 How Can I View My Disk Details?

To do so, perform the following operations:

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 row that contains the target disk and view the disk specifications, attributes, and billing mode.

Step 4 (Optional) Click the disk name to view more information, such as the disk backup and snapshot information.

View more information on the **Summary** tab.

----End

19.2.5 Can I Change the AZ of My Disk?

No.

The AZ of a disk cannot be changed after you have created the disk. If you want to change the AZ, delete the disk and create a new one.

19.2.6 Can I Change the Disk Type, Device Type, or Sharing Attribute of My Disk?

The following table describes whether the disk type, device type, sharing, and encryption attributes of a disk can be changed.

Table 19-1 EVS disk change description

Attribute	Allow Change	Change Direction
Disk type	No	● From high I/O to ultra-high I/O ● From ultra-high I/O to high I/O The preceding descriptions are examples only. EVS disk type cannot be changed.
Sharing	No	● From shared to non-shared ● From non-shared to shared
Device type	No	● From SCSI to VBD ● From VBD to SCSI
Encryption	No	● From non-encrypted to encrypted ● From encrypted to non-encrypted

However, you can:

1. Create a backup for the disk.
2. Create a new disk from this backup. During the creation, select a new disk type and configure advanced settings (sharing, SCSI, and encryption) based on your service requirements.

19.2.7 What Should I Do If an Error Occurs on My EVS Disk?

If an error occurs, the disk may show one of the states listed in the following table. Take the measures described in the table to handle the exceptions.

Table 19-2 Solutions for disk errors

Error Status	Handling Suggestion
Error	Delete the disk in the **Error** state and create another one.
Deletion failed	Contact customer service.

Error Status	Handling Suggestion
Expansion failed	Our customer service personnel will contact you and help you handle this error. Do not perform any operations on the disk before the customer service personnel contact you. If the issue is urgent, you can contact our customer service personnel directly.
Restoration failed	Our customer service personnel will contact you and help you handle this error. Do not perform any operations on the disk before the customer service personnel contact you. If the issue is urgent, you can contact our customer service personnel directly.

19.2.8 How Can I Obtain ECS NIC Information?

Scenarios

This topic shows how to obtain the ECS NIC information, such as the ECS IP address.

Procedure

Step 1 Log in to the management console.

Step 2 Under **Compute**, click **Elastic Cloud Server**.

The ECS console is displayed.

Step 3 In the ECS list, click the target ECS name.

The ECS details page is displayed.

Step 4 Click the **NICs** tab.

The NIC details page is displayed.

Step 5 On the **NICs** tab, click to view ECS NIC information.

Step 6 To view the virtual IP address, click **Manage Virtual IP Address**.

The virtual IP address details page is displayed. You can find the bound virtual IP address based on the ECS NIC information.

----End

19.2.9 Why Do Some of My EVS Disks Not Have WWN Information?

EVS disks have two device types: VBD and SCSI. WWNs are used as the unique identifiers for SCSI EVS disks, and VBD EVS disks do not have WWNs.

You can view the WWN of a SCSI EVS disk on the management console. The details are as follows:

- If the SCSI EVS disk is brand new, you can view the disk WWN on the disk details page.

  Figure 19-1 shows the query result.

  Figure 19-1 Queried WWN information

- If the SCSI EVS disk was created before the WWN feature rollout, the disk WWN will fail to be obtained.

  Figure 19-2 shows the query result.

  Figure 19-2 No WWN information

19.2.10 How Can I Migrate Data from an EVS Disk?

Data migration involves the following scenarios:

- Cross-AZ data migration: Disk data can be migrated from one AZ to another through disk backups. You can create backups for your disks using the CBR service, and then use these backups to create new disks in the target AZ. For details, see sections "Creating a Cloud Disk Backup" and "Using a Backup to Create a Disk" in the *Cloud Backup and Recovery User Guide*.

19.2.11 What Are the Differences Between System Disks and Data Disks?

- A system disk runs the server OS. It is like drive C in a PC.

  When a server is created, a system disk is automatically created and attached. You cannot create a system disk separately. The maximum size of a system disk is 1,024 GiB.

- Data disks store the server data. They are like drive D, drive E, and drive F in a PC.

  Data disks can be created during or after the server creation. If you create data disks during the server creation, the system will automatically attach the data disks to the server. If you create data disks after the server creation, you need to manually attach the data disks. The maximum size of a data disk is 32,768 GiB.

If one system disk already meets your business needs, you do not need to create data disks. As your business grows, you can create data disks when needed.

If the disk paths in your service systems cannot be changed or are difficult to change, you are advised to create data disks according to your system planning.

19.2.12 Will I Lose My Disk Data If I Reinstall ECS OS, Change the OS, or Change the ECS Specifications?

Table 19-3 Impact

Item	OS Reinstallation	OS Change	Specifications Modification
Application scenario	Initialize an ECS. The ECS OS remains unchanged after OS change.	Change the OS of an ECS by changing its image. For details about OS change constraints, see section "Changing the OS" in the *Elastic Cloud Server User Guide*.	Change ECS specifications, such as increasing the number of vCPUs or adding memory, to meet your service requirements.
Billing	OS reinstallation is free of charge. The ECS price remains unchanged.	OS change is free of charge. However, you will be billed based on your new image type after OS change.	Modifying ECS specifications is free of charge. However, you will be billed based on the new specifications after modification.
IP address	The private IP address, EIP, and MAC address remain unchanged.	The private IP address, EIP, and MAC address remain unchanged.	The private IP address, EIP, and MAC address remain unchanged.
System disk	Reinstalling OS will clear the data in all partitions of the ECS system disk. Back up data before reinstalling the OS.	Changing OS will clear the data in all partitions of the ECS system disk. Back up data before changing the OS.	No impact on system disk.
Data disk	No impact on data disk.	No impact on data disk.	No impact on data disk.
Backup	Back up data before reinstalling the OS to prevent data loss.	Back up data before changing the OS to prevent data loss.	Create a system disk snapshot before modifying ECS specifications to prevent data loss.

19.2.13 How Can I Download My EVS Disk Data to a Local PC?

EVS disk data cannot be directly saved to a local PC. It is recommended that you use a third-party tool, such as FTP, to download the data.

19.2.14 How Can I Export the Original Data After I Changed My Server OS from Windows to CentOS?

Solution:

1. Install the ntfsprogs software to enable Linux to access the NTFS file system.
   yum install ntfsprogs
2. View the data disks previously attached to Windows.
   parted –l
3. Mount the data disks.
   mount -t ntfs-3g *Data disk path Mount point*

19.2.15 What Are the Differences Between MBR and GPT Partition Styles?

Table 19-4 lists the common disk partition styles. In Linux, different partition styles require different partitioning tools.

Table 19-4 Disk partition styles

Disk Partition Style	Maximum Disk Capacity Supported	Maximum Number of Partitions Supported	Linux Partitioning Tool
Master Boot Record (MBR)	2 TiB	• 4 primary partitions • 3 primary partitions and 1 extended partition 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: • 3 primary partitions and 1 extended partition, with the extended partition divided into 3 logical partitions • 1 primary partition and 1 extended partition, with the extended partition divided into 5 logical partitions	You can use either of the following tools: • fdisk • parted

Disk Partition Style	Maximum Disk Capacity Supported	Maximum Number of Partitions Supported	Linux Partitioning Tool
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 you change the partition style after the disk has been used, the data on the disk will be cleared. Therefore, select an appropriate partition style when initializing the disk.

19.2.16 What Does the "reserveVolume" Trace Mean in CTS?

Before an EVS disk is attached, the system will call the reserveVolume EVS API to check whether the disk can be attached. If it can be attached, the system then changes the disk status to **attaching** to avoid conflicts with other operations.

19.2.17 How Can I Upload Files to My EVS Disk?

EVS disks must be attached to servers before you can use them. For how to upload files, see section "How Do I Upload Files to My ECS?" in the *Elastic Cloud Server FAQs*.

19.3 Billing

19.3.1 How Can I Stop Being Billed for My Disk?

Pay-per-use disk: Just delete the disk. You will not be billed for a disk after it is deleted. For details, see **Deleting EVS Disks**.

19.3.2 Will I Be Billed If I Have Purchased an EVS Disk But Not Used It?

Yes, you will be billed by disk capacity after you have purchased the disk. The disk billing is irrelevant to whether the disk is attached, how much disk space is used, or whether the attached server is stopped or not.

If your disks are no longer needed, to stop being billed for their use, take the following actions:

Pay-per-use disk: Just delete the disk. You will not be billed for a disk after it is deleted. For details, see **Deleting EVS Disks**.

19.3.3 Can I Recover My Disk Data If the Disk Is Deleted by Mistake?

- Check whether a disk backup is available.
  - If a disk backup is available, you can use it to restore data. If you use a backup to create a disk, the disk will contain the backup data in its initial state. For details, see section "Using a Backup to Create a Disk" in the *Cloud Backup and Recovery User Guide*.

    NOTE

    Disks created from backups do not need to be initialized. Initialization destroys any data that was on the disk.
  - If no disk backup is available, check whether the attached ECS has any backup.
- Check whether the attached ECS has any backup.
  - If a server backup is available, you can use it to restore data. You can restore data of a cloud server backup to the original disk or a different disk. For details, see section "Restoring from a Cloud Server Backup" in the *Cloud Backup and Recovery User Guide*.
  - If neither a cloud disk backup nor a cloud server backup is available, the data cannot be restored.

19.3.4 Will My EVS Disk Be Deleted When I Delete Its Server?

- For pay-per-use disks:
  - If such a disk is separately purchased and has been attached, the system will prompt you whether to delete the disk when you delete the server, and you can make the decision based on your requirements.
  - If such disks are purchased together with a server, the system disk will be deleted when you delete the server. For the data disks, the system will prompt you whether to delete the disks, and you can make the decision based on your service requirements.

19.4 Capacity Expansion

19.4.1 Can I Reduce or Temporarily Expand the Disk Capacity?

No. The disk capacity can only be expanded, and temporary capacity expansion is not supported.

19.4.2 What Are the Differences Between Expanding Capacity by Expanding an EVS Disk and Creating a New EVS Disk?

The differences are as follows:

- Expanding an EVS disk is when you expand the capacity of an existing EVS disk. Some systems let you expand the capacity of EVS disks in use. In this case, services are not interrupted.
- If you create a new EVS disk and attach it to a server that already has an existing EVS disk, the new EVS disk and the original EVS disk are attached to the same server but independent from each other.

19.4.3 Will My Disk Data Be Lost After I Expand the Disk Capacity?

Data will not be deleted during a system disk or data disk capacity expansion. However, incorrect operations during an expansion may result in data loss or exceptions. Exercise caution when performing capacity expansions. You are advised to back up the disk data before expanding capacity. For details about disk backups, see the *Cloud Backup and Recovery User Guide*.

19.4.4 Can I Use Backups Created Before Capacity Expansion to Restore Data on Expanded Disks?

Yes. If backups have been created for disks before capacity was expanded, you can restore your disk data from these backups after the capacity is expanded. Expansion operations do not affect backups.

After the disk data is restored, the disk capacity is increased, but the additional space still needs to be partitioned and formatted before it can be used. You must log in to the server to extend the disk partition or file system.

To extend disk partitions and file systems, see the following sections:

- **Extending Disk Partitions and File Systems (Windows)**
- **Extending Disk Partitions and File Systems (Linux)**

19.4.5 Do I Need to Restart the Server After Expanding the Disk Capacity?

An EVS disk can be expanded either in the Available or In-use state. Expanding the disk capacity on the management console enlarges the disk capacity, but you still need to log in to the server and extend the disk partitions and file systems to make that additional space usable. You may need to restart the server during the partition and file system extension. The details are as follows:

- After expanding an In-use disk on the management console, log in to the server and view the disk capacity.
  - If the additional space can be viewed, you can extend the partition and file system and a restart is not required.
  - If the additional space cannot be viewed, the server OS may not be included in the compatibility list. In this case, you should stop and then

start the server (do not restart the server). When the additional space can be viewed, extend the partition and file system.

- After expanding an Available disk on the management console, attach the disk to the server and extend the partition and file system on the server. In this case, a server restart is not required.

19.4.6 Do I Need to Detach an EVS Disk Before Expanding Its Capacity?

An expansion consists of two phases:

1. Expand the disk capacity on the management console.
   - A shared, in-use disk cannot be expanded. You must detach the shared disk from all its servers and then expand its capacity.
   - A non-shared, in-use disk can be expanded, and you can leave the disk attached during expansion as long as the following conditions are met:
     - The disk's server is in the **Running** or **Stopped** state.
     - The disk's server OS supports the expansion of In-use disks.

       NOTE

       Only some server OSs support capacity expansion of In-use disks. For details, see **Expanding Capacity for an In-use EVS Disk**.
2. Log in to the server and create a new partition or allocate the additional space to one that is already there.
   - In Windows, there are no partition extensions that require the partition to be unmounted first.
   - In Linux:
     - When allocating the additional space to an existing partition, that is, extending an existing partition, you must use the **umount** command to unmount the partition first.
     - When allocating the additional space to a new partition, that is, creating a new partition, you do not need to unmount the existing partition.

19.4.7 What Should I Do If My Disk Capacity Exceeds 2 TiB After Expansion?

An EVS system disk can be as large as 1 TiB (1,024 GiB). You can expand the capacity of a system disk to up to 1 TiB.

An EVS data disk can be as large as 32 TiB (32,768 GiB).

- With MBR, any disk space in excess of 2 TiB cannot be allocated and used, because the maximum disk capacity supported by MBR is 2 TiB (2,048 GiB).

  In this case, if you want to expand the disk capacity to over 2 TiB, change the partition style from MBR to GPT. Ensure that the disk data has been backed up before changing the partition style because services will be interrupted and data on the disk will be deleted during this change.

- With GPT, you can expand the capacity of a data disk to up to 32 TiB because the maximum disk capacity supported by GPT is 18 EiB (19,327,352,832 GiB).

  If the in-use partition style is GPT, see the following methods:
  - Windows:

    Extending Disk Partitions and File Systems (Windows Server 2016)
  - Linux:

    Extending Partitions and File Systems for Data Disks (Linux)

19.4.8 How Can I Allocate Newly Added Space to a New Partition?

Windows

In Windows, if you already have a D drive and want to add an E drive, refer to the following:

The **System Disk: Add Additional Capacity to New Volume (F:)** and **Data Disk: Add Additional Capacity to New Volume (E:)** parts in the following section:

Extending Disk Partitions and File Systems (Windows)

Linux

In Linux, if your disk already has, for example, partition **/dev/vdb1** and you want to create a new partition, **/dev/vdb2**, and allocate the additional space to the new partition, refer to the following:

To extend disk partitions and file systems, see the following sections:

- **Extending Partitions and File Systems for System Disks (Linux)**
- **Extending Partitions and File Systems for Data Disks (Linux)**

19.4.9 How Can I Allocate Newly Added Space to an Existing Partition?

Windows

In Windows, if your disk already has, for example, a D: drive and you want to add space to this volume, refer to the the following:

The **System Disk: Add Additional Capacity to New Volume (C:)** and **Data Disk: Add Additional Capacity to New Volume (D:)** parts in the following section:

Extending Disk Partitions and File Systems (Windows Server 2016)

Linux

In Linux, if your disk already has, for example, partition **/dev/vdb1** and you want to add space to this partition, refer to the following:

To extend disk partitions and file systems, see the following sections:

- **Extending Partitions and File Systems for System Disks (Linux)**

- **Extending Partitions and File Systems for Data Disks (Linux)**

19.4.10 Why Did My Disk Capacity Remain Unchanged on the Server After Capacity Expansion?

After expanding disk capacity on the management console, you must log in to the server and extend the disk partition and file system for the extra capacity to become available. Otherwise, you cannot view the additional space on the server.

To extend disk partitions and file systems, see the following sections:

- **Extending Disk Partitions and File Systems (Windows Server 2008)**
- **Partition and File System Extension Preparations (Linux)**

19.4.11 Why Can't I Expand Capacity for My Disk?

Symptom

Capacity expansion is not allowed for the disk.

Troubleshooting

Possible causes are listed here in order of their probability.

If the fault persists after you have ruled out one cause, move on to the next one in the list.

Table 19-5 Troubleshooting

Possible Cause	Solution
A shared disk is still attached to a server.	See **Shared Disk Is Still Attached to a Server**.

Shared Disk Is Still Attached to a Server

Symptom: The **Expand Capacity** button is grayed out. When you attempt to click the capacity expansion button, the following hover tip is displayed: This operation can be performed only when the shared disk is in the Available state.

Solution: Detach the disk from all servers. If the **Expand Capacity** button becomes available, you can expand the disk capacity.

19.4.12 How Do I Extend the File System of an Unpartitioned Data Disk in Linux?

Scenarios

If no partition but only a file system is created on a data disk, extend the file system according to the following operations:

Run the **lsblk** command. Information similar to the following is displayed:

```
[root@ecs-test ~]# 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   60G  0 disk /mnt/sdc
```

In the command output, no partition but only a file system is created on data disk **/dev/vdb**.

In the following example, CentOS 7.4 64bit is used as the sample OS, data disk **/dev/vdb** has 10 GiB, no partition but only a file system is created on the disk, and additional 50 GiB has been added to this data disk on the management console. The following steps show how to extend this 50 GiB to the file system.

- **Extending the EXT* File System**
- **Extending the XFS File System**

The way you allocate additional space depends on the OS. This example is used for reference only. For the detailed operations and differences, see the corresponding OS documentations.

Extending the EXT* File System

Step 1 Run the following command to extend the file system:

resize2fs *Disk name*

In this example, run the following command:

resize2fs /dev/vdb

Information similar to the following is displayed:

```
[root@ecs-test ~]# resize2fs /dev/vdb
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/vdb is mounted on /root/test; on-line resizing required
old_desc_blocs = 2, old_desc_blocs = 8
[17744.521535] EXT4-fs (vdb): resizing filesystem from 26214400 to 15728640 blocks
[17744.904470] EXT4-fs (vdb): resized filesystem to 15728640
The filesystem on /dev/vdb is now 15728640 blocks long.
```

Step 2 Run the following command to view the result:

df -TH

Information similar to the following is displayed:

```
[root@ecs-test ~]# 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/vdb       ext4       64G   55M   61G   1% /mnt/sdc
```

----End

Extending the XFS File System

Step 1 Run the following command to extend the file system:

xfs_growfs *Disk name*

In this example, run the following command:

xfs_growfs /dev/vdb

Information similar to the following is displayed:

```
[root@ecs-test ~]# xfs_growfs /dev/vdb
meta-data=/dev/vdb               isize=512    agcount=4, agsize=655360 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, spinodes=0
data     =                       bsize=4096   blocks=2621440, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version2               bsize=4096   ascii-ci=0 ftype=1
log      =internal               bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 2621440 to 15728640.
```

Step 2 Run the following command to view the result:

df -TH

Information similar to the following is displayed:

```
[root@ecs-test ~]# df -TH
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/vda1      ext4       40G  2.3G   35G   7% /
devtmpfs       devtmpfs  1.9G     0  1.9G   0% /dev
tmpfs          tmpfs     1.9G     0  1.9G   0% /dev/shm
tmpfs          tmpfs     1.9G  8.6M  1.9G   1% /run
tmpfs          tmpfs     1.9G     0  1.9G   0% /sys/fs/cgroup
tmpfs          tmpfs     379M     0  379M   0% /run/user/0
/dev/vdb       xfs       60G   34M   60G   1% /mnt/sdc
```

----End

19.4.13 How Do I Extend the Root Partition of a Quickly Provisioned BMS?

Scenarios

If the root partition of your quickly provisioned BMS is too small, extend the root partition by referring to the following procedure.

This example uses CentOS 7.3 and system disk, **/dev/sdf**. The way you allocate additional space depends on the OS. This example is used for reference only. For detailed operations and differences, see the corresponding OS documentations.

In this example, the initial size of the BMS system disk (**sdf**) is 40 GiB and needs to be expanded to 140 GiB. The initial disk partitions are as follows:

```
sdf        8:80   0    40G  0 disk
├─sdf1     8:81   0   500M  0 part /boot
├─sdf2     8:82   0     5G  0 part [SWAP]
├─sdf3     8:83   0  34.5G  0 part /
└─sdf4     8:84   0    64M  0 part
```

Procedure

Step 1 Log in to the EVS console and expand the system disk capacity to 140 GiB.

Step 2 Log in to the BMS and run the following command to view the system disk capacity:

lsblk

Information similar to the following is displayed:

```
sdf       8:80   0   140G  0 disk
├─sdf1    8:81   0   500M  0 part /boot
├─sdf2    8:82   0     5G  0 part [SWAP]
├─sdf3    8:83   0  34.5G  0 part /
└─sdf4    8:84   0    64M  0 part
```

The system disk (**sdf**) has been expanded from 40 GiB to 140 GiB. The **sdf4** partition (64 MiB) is the configdriver partition that stores the BMS configuration information.

Step 3 Run the following command to back up the configdriver partition:

dd if=/dev/sdf4 of=/root/configdriver.img

Information similar to the following is displayed:

```
[root@bms-6acd ~]# dd if=/dev/sdf4 of=/root/configdriver.img
131072+0 records in
131072+0 records out
67108864 bytes (67 MB) copied, 0.291739 s, 230 MB/s
```

Step 4 Run the following command and delete the configdriver partition:

fdisk /dev/sdf

```
[root@bms-6acd ~]# fdisk /dev/sdf
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.


Command (m for help): d
Partition number (1-4, default 4): 4
Partition 4 is deleted

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

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource
 busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

Step 5 Run the **partprobe** command to update the partition information.

If partition configdriver has been deleted, information similar to the following is displayed:

```
sdf        8:80   0   140G  0 disk
├─sdf1     8:81   0   500M  0 part /boot
├─sdf2     8:82   0     5G  0 part [SWAP]
├─sdf3     8:83   0  34.5G  0 part /
└─sdf4     8:84   0  97.7M  0 part
```

Step 6 Recreate the configdriver partition with 100 MB.

If the available sectors range from 83755008 to 293601279, set 293401279 (293601279 – 200000) as the new partition's start sector and 293601279 (default value) as the end sector.

```
Command (m for help): n
Partition type:
   p   primary (3 primary, 0 extended, 1 free)
   e   extended
Select (default e): p
Selected partition 4
First sector (83755008-293601279, default 83755008): 293401279
Last sector, +sectors or +size{K,M,G} (293401279-293601279, default 293601279):
Using default value 293601279
Partition 4 of type Linux and of size 97.7 MiB is set

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

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource
 busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

Run the **partprobe** command to update the partition information.

```
sdf        8:80   0   140G  0 disk
├─sdf1     8:81   0   500M  0 part /boot
├─sdf2     8:82   0     5G  0 part [SWAP]
├─sdf3     8:83   0  34.5G  0 part /
└─sdf4     8:84   0  97.7M  0 part
```

Step 7 Run the following command to extend the root partition:

growpart /dev/sdf 3

Information similar to the following is displayed:

```
[root@bms-6acd ~]# growpart /dev/sdf 3
CHANGED: partition=3 start=11511808 old: size=72243200 end=83755008 new:size=28
1889471,end=293401279
```

Run the **lsblk** command to view the new root partition size.

```
sdf        8:80   0   140G  0 disk
├─sdf1     8:81   0   500M  0 part /boot
├─sdf2     8:82   0     5G  0 part [SWAP]
├─sdf3     8:83   0 134.4G  0 part /
└─sdf4     8:84   0  97.7M  0 part
```

Step 8 Run the following command to extend the file system of the root partition:

resize2fs /dev/sdf3

Information similar to the following is displayed:

```
[root@bms-6acd ~]# resize2fs /dev/sdf3
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/sdf3 is mounted on /; on-line resizing required
old_desc_blocks = 5, new_desc_blocks = 17
The filesystem on /dev/sdf3 is now 35236183 blocks long.

You have new mail in /var/spool/mail/root
[root@bms-6acd ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sdf3       133G  1.9G  125G   2% /
devtmpfs         63G     0   63G   0% /dev
tmpfs            63G     0   63G   0% /dev/shm
tmpfs            63G  9.0M   63G   1% /run
tmpfs            63G     0   63G   0% /sys/fs/cgroup
/dev/sdf1       477M  104M  344M  24% /boot
tmpfs            13G     0   13G   0% /run/user/0
```

Step 9 Run the following command to restore the content of the configdriver partition:

dd if=/root/configdriver.img of=/dev/sdf4

Information similar to the following is displayed:

```
[root@bms-6acd ~]# dd if=/root/configdriver.img of=/dev/sdf4
131072+0 records in
131072+0 records out
67108864 bytes (67 MB) copied, 0.372614 s, 180 MB/s
[root@bms-6acd ~]# _
```

```
/dev/sdf1: UUID="b9c472f9-6737-4200-910a-efa3af16629a" TYPE="ext4"
/dev/sdf2: UUID="b07ff4d0-8b0b-4c43-a40a-0b27290ea215" TYPE="swap"
/dev/sdf3: UUID="1e57f71e-6adc-4e98-9407-0f7d678d4525" TYPE="ext4"
/dev/sdf4: UUID="2018-09-27-19-13-01-00" LABEL="config-2" TYPE="iso9660"
[root@bms-6acd ~]# _
```

The root partition of the quickly provisioned BMS has been extended.

----End

19.4.14 How Do I View the Disk Partition Style in Linux?

You can use either fdisk or parted to view the disk partition style.

- **Method 1: Check Partition Style and File System Format Using fdisk**
- **Method 2: Check Partition Style and File System Format Using parted**

Method 1: Check Partition Style and File System Format Using fdisk

Step 1 Run the following command to view all the disks attached to the server:

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  150G  0 disk
└─vdb1 253:17   0  100G  0 part /mnt/sdc
```

In this example, data disk **/dev/vdb** already has partition **/dev/vdb1** before capacity expansion, and the additional 50 GiB added has not been allocated yet. Therefore, **/dev/vdb** has 150 GiB, and **/dev/vdb1** has 100 GiB.

NOTE

If you run **lsblk** and find out that disk **/dev/vdb** has no partitions, format the disk by referring to **How Do I Extend the File System of an Unpartitioned Data Disk in Linux?** and expand the capacity. Otherwise, the additional space cannot be used after expansion.

Step 2 Run the following command to view the current disk partition style:

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: 161.1 GiB, 161061273600 bytes, 314572800 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
```

The value in the **System** column indicates the disk partition style. Value **Linux** indicates the MBR partition style. Value **GPT** indicates the GPT partition style.

- If the disk partitions displayed are inconsistent with those obtained in **Step 1**, the possible reason may be that existing partitions uses GPT and there is unallocated disk space. In this case, you cannot query all the partitions using **fdisk -l**. Go to **Method 2: Check Partition Style and File System Format Using parted**.
- If the disk partitions displayed are consistent with those obtained in **Step 1**, continue with the following operations.

Step 3 Run the following command to view the partition's file system format:

blkid *Disk partition*

In this example, run the following command:

blkid /dev/vdb1

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

In the command output, the **TYPE** value is **ext4**, indicating that **/dev/vdb1**'s file system format is **ext4**.

Step 4 Run the following command to view the file system status:

ext*: **e2fsck -n** *Disk partition*

xfs: **xfs_repair -n** *Disk partition*

In this example, the ext4 file system is used. Therefore, run the following command:

e2fsck -n /dev/vdb1

Information similar to the following is displayed:

```
[root@ecs-test-0001 ~]# e2fsck -n /dev/vdb1
e2fsck 1.42.9 (28-Dec-2013)
Warning!  /dev/vdb1 is mounted.
Warning: skipping journal recovery because doing a read-only filesystem check.
/dev/vdb1: clean, 11/6553600 files, 459544/26214144 blocks
```

If the file system status is **clean**, the file system is normal. Otherwise, rectify the faulty and then perform the capacity expansion.

----End

Method 2: Check Partition Style and File System Format Using parted

Step 1 Run the following command to view all the disks attached to the server:

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  150G  0 disk
└─vdb1 253:17   0  100G  0 part /mnt/sdc
```

In this example, data disk **/dev/vdb** already has partition **/dev/vdb1** before capacity expansion, and the additional 50 GiB added has not been allocated yet. Therefore, **/dev/vdb** has 150 GiB, and **/dev/vdb1** has 100 GiB.

NOTE

If you run **lsblk** and find out that disk **/dev/vdb** has no partitions, format the disk by referring to **How Do I Extend the File System of an Unpartitioned Data Disk in Linux?** and expand the capacity. Otherwise, the additional space cannot be used after expansion.

Step 2 Run the following command and enter **p** to view the disk partition style:

parted *Disk*

For example, run the following command to view **/dev/vdb**'s partition style:

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) p
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another
operating system believes the
disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of
```

```
the space (an extra 104857600
blocks) or continue with the current setting?
Fix/Ignore? Fix
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 161GiB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End    Size   File system  Name  Flags
 1      1049kB  107GiB  107GiB  ext4         test

(parted)
```

Partition Table indicates the disk partition style. **Partition Table: msdos** means MBR, **Partition Table: gpt** means GPT, and **Partition Table: loop** means that the whole disk is partitioned.

- If the following error information is displayed, enter **Fix**.
  ```
  Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another operating system believes the
  disk is smaller.  Fix, by moving the backup to the end (and removing the old backup)?
  ```
  The GPT partition table information is stored at the start of the disk. To reduce the risk of damage, a backup of the information is saved at the end of the disk. When you expand the disk capacity, the end of the disk changes accordingly. In this case, enter **Fix** to move the backup file of the information to new disk end.
- If the following warning information is displayed, enter **Fix**.
  ```
  Warning: Not all of the space available to /dev/vdb appears to be used, you can fix the GPT to use all of the space (an extra 104857600
  blocks) or continue with the current setting?
  Fix/Ignore? Fix
  ```
  Enter **Fix** as prompted. The system automatically sets the GPT partition style for the additional space.

Step 3 Enter **q** and press **Enter** to exit parted.

----End

19.5 Attachment

19.5.1 Why Can't I View the Attached Data Disk on the Server?

Troubleshooting

Table 19-6 Possible causes

OS	Possible Cause	Solution
Linux	• New data disks are not formatted and partitioned by default, and an unformatted disk will not be listed in the command output. You must manually initialize the disk. • If a data disk cannot be found after the server is restarted, automatic partition mounting at system start may not be configured.	**Linux Data Disk**
Windows	New data disks are not formatted and partitioned by default. Only formatted and partitioned drives show up in the resource manager. You must manually initialize the disk.	**Windows Data Disk**

Linux Data Disk

Symptom: A data disk has been attached to a Linux server on the management console, but the disk cannot be viewed on the server.

Run **df -TH** to view the disk information. CentOS 7.4 is used in this example. The normal command output is as follows:

```
[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
```

Unlike the normal command output, only system disk **/dev/vda1** is visible, but data disk **/dev/vdb1** is missing from the command output.

Cause Analysis:

- **Cause 1**: New data disks are not formatted and partitioned by default, and an unformatted disk will not be listed in the command output. You must manually initialize the disk.

  For details, see **Introduction to Data Disk Initialization Scenarios and Partition Styles**.

- **Cause 2**: If a data disk cannot be found after the server is restarted, automatic partition mounting at system start may not be configured. Perform the following steps:

a. Run the following command to mount the partition again:

mount *Disk partition Mount point*

In this example, run the following command:

mount /dev/vdb1 /mnt/sdc

Perform the following steps to enable automatic partition mounting at system start:

b. Run the following command to query the partition UUID:

blkid *Disk partition*

In this example, run the following command to query the UUID of the **/dev/vdb1** partition:

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"
```

The UUID of the **/dev/vdb1** partition is displayed.

c. Run the following command to open the **fstab** file using the vi editor:

vi /etc/fstab

Press **i** to enter editing mode.

d. 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
```

The preceding content is used for reference only. Add the information that is used in the environment. The parameters are described as follows:

- The first column indicates the partition UUID obtained in **b**.
- The second column indicates the directory on which the partition is mounted. You can query the mount point using the **df -TH** command.
- The third column indicates the file system format of the partition. You can query the file system format using the **df -TH** command.
- The fourth column indicates the partition mount option. Normally, this parameter is set to **defaults**.
- The fifth column indicates the Linux dump backup option.
  - **0**: Linux dump backup is not used. Normally, dump backup is not used, and you can set this parameter to **0**.
  - **1**: Linux dump backup is used.
- The sixth column indicates the fsck option, that is, whether to use fsck to check the attached disk during startup.
  - **0**: not use fsck.
  - If the mount point is the root partition (**/**), this parameter must be set to **1**.

    When 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.

e. Press **Esc**, enter **:wq**, and press **Enter**.

The system saves the configurations and exits the vi editor.

Perform the following operations to verify the automatic mounting function:

i. Run the following command to unmount the partition:

umount *Disk partition*

In this example, run the following command:

umount /dev/vdb1

ii. Run the following command to reload all the content in the **/etc/fstab** file:

mount -a

iii. Run the following command to query the file system mounting information:

mount | grep *Mount point*

In this example, run the following command:

mount | grep /mnt/sdc

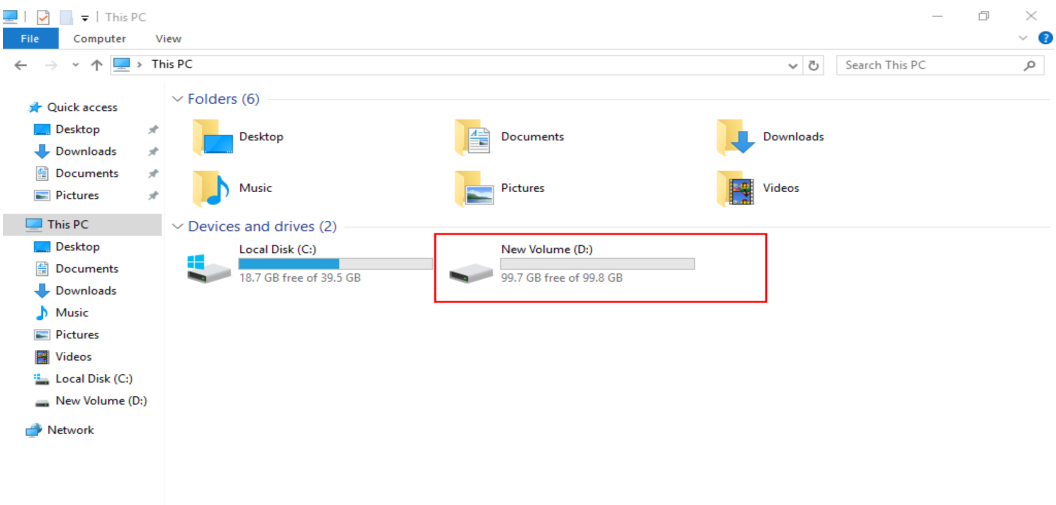
If information similar to the following is displayed, automatic mounting has been configured:

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

Windows Data Disk

Symptom: A data disk has been attached to a Windows server on the management console, but the disk cannot be viewed on the server. For example, Volume (D:) was not shown in **This PC** of a Windows server running Windows Server 2012. Normally, Volume (D:) appears, as shown in **Figure 19-3**.

Figure 19-3 Volume (D:) appears



Solution: New data disks are not formatted and partitioned by default. Only formatted and partitioned drives show up in **This PC**. You must manually initialize the disk before it can be viewed here.

For details, see **Introduction to Data Disk Initialization Scenarios and Partition Styles**.

19.5.2 Why Can't I Attach My Disk to a Server?

Symptom

My disk cannot be attached to a server.

Troubleshooting

Possible causes are listed here in order of their probability.

If the fault persists after you have ruled out one cause, move on to the next one in the list.

Table 19-7 Troubleshooting

Symptom	Solution
The target server on the **Attach Disk** page could not be found.	• Go to **Check Whether the Disk and Server Are in the Same AZ**. • Cloud servers created from ISO images are only used for OS installation. They have limited functions and cannot have EVS disks attached.
The **Attach** button is grayed out.	• Go to **Maximum Number of Disks That Can Be Attached to the Server Has Been Reached**. • Go to **Check Whether the Disk Has Been Added to a Replication Pair**.
An incorrect OS type warning is displayed when a shared disk is attached.	Go to **Check Whether the Servers Attached with the Shared Disk Are Running the Same Type of OS**.

Check Whether the Disk and Server Are in the Same AZ

Symptom: After you click **Attach**, the target server cannot be found on the **Attach Disk** page.

Solution: A disk can only be attached to a server in the same AZ and region. The **Attach Disk** page filters and shows all the servers that the disk can be attached to. Determine whether your disk data is required.

- If the disk data is no longer needed, delete the disk, and then create a new disk in the AZ where your target server is located.
- If the disk data is still required, create a new disk with the same data in the target AZ. The procedure is as follows:

a. Create a backup for the disk.
b. Create a new disk from this backup. During the creation, select the target AZ. You can also change the settings of **Disk Type** and **Advanced Settings** if needed.
c. After the disk is created, click **Attach**. Your target server is displayed on the **Attach Disk** page.

Maximum Number of Disks That Can Be Attached to the Server Has Been Reached

Symptom: The **Attach** button is grayed out.

Solution:

- Non-shared disk: When you hover the mouse over the **Attach** button, message "This operation can be performed only when the disk is in the Available state" is displayed.

  A non-shared disk can only be attached to one server. If the disk status is **In-use**, the disk has been attached. You can detach the disk, wait until the disk status changes to **Available**, and then attach the disk to the target server.
- Shared disk: When you hover the mouse over the **Attach** button, message "This operation cannot be performed because the maximum number of servers that a shared disk can be attached to has been reached" is displayed.

  A shared disk can be attached to a maximum of 16 servers, but you can detach the shared disk from one server and attach it to a new one if needed.

Check Whether the Disk Has Been Added to a Replication Pair

Symptom: The **Attach** button is grayed out. When you hover the mouse over the **Attach** button, message "This operation cannot be performed on a disk in a replication pair" is displayed.

Solution: Delete the replication pair and attach the disk again.

Step 1 Choose **Storage** > **Storage Disaster Recovery Service**.

The **Storage Disaster Recovery Service** page is displayed.

Step 2 Locate the protection group containing the disk and click the protection group name.

The protection group details page is displayed.

Step 3 Click the **Replication Pairs** tab.

Check that the disk in the **Production Site Disk** column is the target disk.

Step 4 Confirm the information and click **Delete** in the **Operation** column.

Step 5 After the replication pair is deleted, return to the disk list, and the disk can be attached.

----End

Check Whether the Servers Attached with the Shared Disk Are Running the Same Type of OS

Symptom: After you click **Attach**, the target server cannot be selected on the **Attach Disk** page, and message "A shared disk must be attached to servers with the same OS type" is displayed.

Solution: This message indicates that the OS type of the target server is inconsistent with that of the servers attached with the shared disk. You can change the OS type based your service requirements.

19.5.3 Can I Attach a Disk to Multiple Servers?

- A non-shared disk can only be attached to one server.
- A shared disk can be attached to up to 16 servers.

NOTE

- Shared disks are a type of EVS disks that can be attached to multiple servers.
- To use shared disks, you must set up a shared file system or cluster management system. If you directly attach a disk to multiple servers, the disk sharing attribute does not work and data may be overwritten.

19.5.4 Can I Attach a Disk to a Server in Another AZ?

No.

Disks and the servers you attach the disks to must be in the same AZ. The same is true for shared disks. Shared disks can only be attached to the servers in the same AZ.

19.5.5 How Can I Add a Data Disk to an Existing Server?

Data disks can be created during or after the server creation. If you create data disks during the server creation, the system will automatically attach the data disks to the server. If you create data disks after the server creation, you need to manually attach the data disks.

- On a Windows server:
  - If a data disk is created along with the server, you need to log in to the server and initialize the disk. The data disk will be visible after the initialization succeeds.
  - If no data disk is created along with the server, you need to create a data disk and attach it to the server. Then, you need to log in to the server and initialize the disk. The data disk will be visible after the initialization succeeds.
- On a Linux server:
  - If a data disk is created along with the server, you need to log in to the server and initialize the disk. The data disk will be visible after the initialization succeeds and the disk has been mounted via the **mount** command.
  - If no data disk is created along with the server, you need to create a data disk and attach it to the server. Then, you need to log in to the server

and initialize the disk. The data disk will be visible after the initialization succeeds and the disk has been mounted via the **mount** command.

For details, see **Introduction to Data Disk Initialization Scenarios and Partition Styles**.

19.5.6 Can I Attach Different Types of Disks to the Same Server?

Yes. Different types of EVS disks can be attached to the same server. You only need to make sure that these disks and the server are in the same AZ.

19.5.7 What Should I Do If a Linux EVS Disk Is Attached to a Windows Server?

You are not advised to attach a Linux EVS disk to a Windows server or attach a Windows EVS disk to a Linux server.

The disk information may fail to be displayed due to the inconsistent file systems. To solve this issue, you need to initialize and partition the disk again. Formatting the disk will destroy any data the disk has contained, so you should back up the data first.

19.5.8 Can I Change the Function of a System Disk or Data Disk Created Along with a Server?

This section describes how to change the function of a disk on the original server.

- System disk created along with a server: You can detach the system disk and then re-attach it to the same server as a system disk or data disk.

  NOTE

  Each server can only have one system disk. If there is already a system disk attached to the server, then the detached system disk cannot be attached as a second system disk.

- Data disk created along with a server: You can detach the data disk and then re-attach it to the same server only as a data disk. It cannot be attached as a system disk.

19.5.9 How Do I Obtain My Disk Device Name in the ECS OS Using the Device Identifier Provided on the Console?

Scenarios

You find that the device name displayed in the ECS OS is different from that displayed on the management console and you cannot determine which disk name is correct. This section describes how to obtain the disk name used in an ECS OS according to the device identifier on the console.

For details about how to attach disks, see **Attach an EVS Disk**.

Using a Serial Number to Obtain the Disk Name (Windows)

If a serial number is displayed on the console, use either of the following methods to obtain the disk name.

cmd

1. Start **cmd** in a Windows OS as an administrator and run either of the following commands:

   wmic diskdrive get serialnumber

   wmic path win32_physicalmedia get SerialNumber

   wmic path Win32_DiskDrive get SerialNumber

   NOTE

   A serial number is the first 20 digits of a disk UUID.

   For example, if the serial number of a VBD disk on the console is 97c876c0-54b3-460a-b, run either of the following commands to obtain the serial number of the disk on the ECS OS:

   wmic diskdrive get serialnumber

   wmic path win32_physicalmedia get SerialNumber

   wmic path Win32_DiskDrive get SerialNumber

   Information similar to the following is displayed:

   Figure 19-4 Obtaining the disk serial number

```
C:\Users\Administrator>wmic diskdrive get serialnumber
SerialNumber
97c876c0-54b3-460a-b


C:\Users\Administrator>wmic path win32_physicalmedia get SerialNumber
SerialNumber
97c876c0-54b3-460a-b


C:\Users\Administrator>wmic path Win32_DiskDrive get SerialNumber
SerialNumber
97c876c0-54b3-460a-b
```

2. Run the following command to check the disk corresponding to the serial number:

   wmic diskdrive get Name, SerialNumber

   Figure 19-5 Checking the disk corresponding to the serial number

```
C:\Users\Administrator>wmic diskdrive get Name, SerialNumber
Name                    SerialNumber
\\.\PHYSICALDRIVE0      97c876c0-54b3-460a-b
```

PowerShell

1. Start PowerShell as an administrator in a Windows OS.
2. Run the following command to check the disk on which the logical disk is created:

- Windows Server 2012 or later
  i. Run the following command to check the disk on which the logical disk is created:

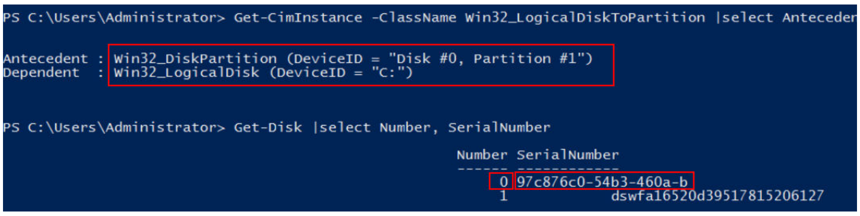
     Get-CimInstance -ClassName Win32_LogicalDiskToPartition | select Antecedent, Dependent |fl

     As shown in **Figure 19-6**, the disk is **Disk 0**.
  ii. Run the following command to view the mapping between the serial number and the disk:

     Get-Disk |select Number, SerialNumber

     As shown in **Figure 19-6**, the disk is **Disk 0**.

     Figure 19-6 Viewing the disk on which the logical disk is created

- Versions earlier than Windows 2012
  i. Run the following command to check the disk on which the logical disk is created:

     Get-WmiObject -Class Win32_PhysicalMedia |select Tag, Serialnumber
  ii. Run the following command to view the mapping between the serial number and the disk:

     Get-WmiObject -Class Win32_LogicalDiskToPartition |select Antecedent, Dependent |fl

Background

Disk information displayed varies according to the ECS virtualization type. For the sake of convenience, ECSs that use the KVM virtualization type are called KVM instances, and ECSs that use the Xen virtualization type are called Xen instances.

Obtaining the Disk Device Name of a KVM Instance

Step 1 Obtain the disk information displayed on the console.

1. Log in to the management console.
2. Under **Computing**, click **Elastic Cloud Server**.
3. Click the target ECS name in the ECS list.

   The page providing details about the ECS is displayed.
4. Click the **Disks** tab and then to expand the disk information.
5. Check the device type and ID of the disk.
   - If the device type is **VBD**, go to **Step 2**.

– If the device name is **SCSI**, go to **Step 3**.

NOTE

If **Device Identifier** is not displayed on the web page, stop the ECS and restart it.

Step 2 Check the device name of a VBD disk attached to the ECS.

1. Obtain the disk device ID by referring to **Step 1**.

   The device ID of the VBD disk shows the PCI address of the disk on the ECS. The address is in the format of "domain:bus:slot.function".

2. Log in to the ECS as user **root**.
3. In **/sys/bus/pci/devices/DOMIN:BUS:SLOT.FUNCTION/virtio*/block**, view the device name.

   For example, if the device ID of the VBD disk is **0000:00:05.0**, the device name is shown as follows:

```
A90CF6C6-BEC0-0C44-8082-8C8610755B61:/sys/bus/pci/devices/0000:00:05.0/virtiol/block #
ll /sys/bus/pci/devices/0000:00:05.0/virtio1/block total 0
drwxr-xr-x 10 root root 0 May 22 11:01 vda
```

   The displayed information is the disk device name, **/dev/vda** in the preceding figure.

Step 3 Check the device name of a SCSI disk attached to the ECS.

1. Obtain the disk device ID by referring to **Step 1**.

   The device ID of the SCSI disk displays the disk WWN on the ECS.

2. Log in to the ECS as user **root**.
3. Run the following command to view the disk device name:

   ll /dev/disk/by-id |grep *WWN*|grep scsi-3

```
[root@host-192-168-133-148 block]# ll /dev/disk/by-id/ |grep 6888603000008b32fa16688d09368506 |
grep scsi-3
lrwxrwxrwx 1 root root  9 May 21 20:22 scsi-36888603000008b32fa16688d09368506 -> ../../sda
```

----End

Obtaining the Disk Device Name of a Xen Instance

Step 1 Obtain the disk information displayed on the console.

1. Log in to the management console.
2. Under **Computing**, choose **Elastic Cloud Server**.
3. Click the target ECS name in the ECS list.

   The ECS details page is displayed.

4. Click the **Disks** tab and then to expand the disk information.
5. Check the device name, type, and ID of the disk.
   - If the device type is **VBD**, go to **Step 2**.
   - If the device name is **SCSI**, go to **Step 3**.

NOTE

If **Device Identifier** is not displayed on the page, stop the ECS and restart it.

Step 2 Check the device name attached to the VBD disk in the ECS.

For a VBD disk, the device name displayed on the management console corresponds to the disk device name in the ECS OS. For details, see ECS.

Table 19-8 Mapping between disk device names displayed on the management console and those obtained on the ECS

Device Name (on Management Console)	Device Name (in ECS)
/dev/sd***	/dev/xvd***
/dev/vd***	/dev/xvd***
/dev/xvd***	/dev/xvd***

An example is provided as follows:

If the device name displayed on the management console is **/dev/sdb**, the device name of the device attached to the ECS is **/dev/xvdb**.

Step 3 Check the device name of the SCSI disk attached to the ECS.

1. Obtain the disk device ID.

   The device ID of the SCSI disk is the disk WWN on the ECS.
2. Log in as user root.
3. Run the following command to view the disk device name:

   ll /dev/disk/by-id |grep *WWN*|grep scsi-3

   ```
   [root@host-192-168-133-148 block]# ll /dev/disk/by-id/ |grep 6888603000008b32fa16688d09368506 |
   grep scsi-3
   lrwxrwxrwx 1 root root  9 May 21 20:22 scsi-36888603000008b32fa16688d09368506 -> ../../sda
   ```

----End

19.6 Detachment

19.6.1 If I Detach a Disk, Will I Lose the Data on My Disk?

It depends on if the disk is encrypted or not.

- Encrypted
  - The CMK is disabled or scheduled for deletion.

    The disk can still be used, but there is no guarantee for how long it will be usable. If the disk is detached, it will be impossible to re-attach it later. In this case, do not detach the disk without a working CMK.

    The restoration method varies depending on the CMK status. For details, see **EVS Encryption**.
  - The CMK is available.

The disk can be detached and re-attached, and data on the disk will not be lost.

To ensure your data safety, you are advised to follow the instructions described in **Disk Detachment Process**.

- Non-encrypted

  Data on a disk will not be lost after the disk is detached, and the disk can be re-attached later if needed.

  To ensure your data safety, you are advised to follow the instructions described in **Disk Detachment Process**.

Disk Detachment Process

- For disks not supporting online detachment:
  a. Stop the server where the disk was attached.
  b. After the server has stopped, detach the disk.
- For disks supporting online detachment:

  Detach the disk from a running ECS. For details, see **Management** > **Detaching an EVS Disk from a Running ECS** in the *Elastic Cloud Server User Guide*.

19.6.2 Why Can't I Detach My Disk?

EVS disks can be used as system disks or data disks, but the way you detach each one is different.

- System disks: A system disk can only be detached offline. You must first stop the server that uses this system disk and then detach the disk.

  NOTE

  In Linux, a system disk is typically mounted on **/dev/vda**. In Windows, a system disk is normally **Volume (C:)**.

- Data disks: A data disk can be detached regardless of whether it is offline or online.

  NOTE

  In Linux, a data disk is typically mounted on a mount point other than **/dev/vda**. In Windows, a data disk is normally a volume other than **Volume (C:)**.

  – Offline detachment: The server must be in the **Stopped** state. If it is not, stop the server and then detach the data disk.
  – Online detachment: Some OSs support online detachment. In this case, you do not need to stop the server before detaching the data disk. For more information, see **Storage** > **Detaching an EVS Disk from a Running ECS** in the *Elastic Cloud Server User Guide*.

19.7 Deletion

19.7.1 How Can I Recover Data from a Disk That Was Accidentally Deleted?

Check whether the disk has any backups created.

- If there are, use a backup to restore the disk data to the state when the backup was created. For details, see section "Restoring from a Cloud Disk Backup" in the *Cloud Backup and Recovery User Guide*.

  NOTICE

  If the disk was deleted after the last backup was created, that incremental data cannot be restored.

- If there are not, the disk data cannot be restored.

19.8 Capacity

19.8.1 What Is the Maximum Capacity Supported for the System and Data Disks?

The maximum capacity supported for a system disk is 1024 GiB.

The maximum capacity supported for a data disk is 32768 GiB.

19.8.2 What Should I Do If My Disk Starts to Run Out of Space?

If your disk space starts to fill up, you can:

1. Create a new disk and attach it to the server. For details, see **Create an EVS Disk**.
2. Expand the capacity of the existing disk. Both system disks and data disks can be expanded. An expansion operation includes two steps:
   a. Expand the disk capacity on the console.
   b. Log in to the server and extend the partition and file system.

   For details, see **Disk Capacity Expansion**

Differences Between Expanding an EVS Disk and Creating an EVS Disk

The differences are as follows:

- Expanding an EVS disk is when you expand the capacity of an existing EVS disk. Some systems let you expand the capacity of EVS disks in use. In this case, services are not interrupted.
- If you create a new EVS disk and attach it to a server that already has an existing EVS disk, the new EVS disk and the original EVS disk are attached to the same server but independent from each other.

19.8.3 What Can I Do If the Capacity of My Disk Reaches the Maximum But I Still Need More Space?

The capacity of a single disk can be expanded to up to 32 TiB. If this still fails to meet your needs, it is recommended that you create RAID arrays with EVS disks or manage EVS disks using LVM.

19.8.4 What Should I Do If I Use fdisk to Initialize a Disk Larger Than 2 TiB and Then the Space in Excess of 2 TiB Cannot Be Displayed?

If your disk capacity is greater than 2 TiB, do not use fdisk to partition the disk. Or any space in excess of 2 TiB will be unable to show up after the disk is partitioned.

In this case, use parted to repartition the disk and choose the GPT partition style because MBR does not support disks over 2 TiB.

For details, see **Introduction to Data Disk Initialization Scenarios and Partition Styles**.

19.8.5 How Can I View My Disk Usage?

You can view your disk usages in either of the following ways:

- View disk usages manually.

  The details depend on the OS. This FAQ uses Windows Server 2008, Windows Server 2016, and Linux as samples to describe how to view the disk usage.
  - **Viewing Disk Usage in Linux**
  - **Viewing Disk Usage in Windows Server 2008**
  - **Viewing Disk Usage in Windows Server 2016**
- **Installing Agent to View Disk Usage**

Viewing Disk Usage in Linux

In this section, CentOS 7.4 64bit is used as an example. The details depend on if you need to view the available space or not.

- To query the total capacity only, run **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   40G  0 disk
└─vdb1 253:17   0   40G  0 part
```
  In the command output, the server has two disks, **/dev/vda** and **/dev/vdb**. System disk **/dev/vda** has 40 GiB of capacity, as does data disk **/dev/vdb**.
- To query the total capacity and display the space available as well, run **df -TH**. Ensure that the disk has been attached and initialized before running this command.

  Information similar to the following is displayed:

```
[root@ecs-0001 ~]# df -TH
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/vda1      ext4       43G  2.0G   39G   5% /
devtmpfs       devtmpfs  509M     0  509M   0% /dev
tmpfs          tmpfs     520M     0  520M   0% /dev/shm
tmpfs          tmpfs     520M  7.2M  513M   2% /run
tmpfs          tmpfs     520M     0  520M   0% /sys/fs/cgroup
tmpfs          tmpfs     104M     0  104M   0% /run/user/0
/dev/vdb1      ext4       43G   51M   40G   1% /mnt/sdc
```

In the command output, the server has two partitions, **/dev/vda1** and **/dev/vdb1**. Partition **/dev/vda1** is used to deploy the OS, and its total capacity, used capacity, and available capacity are 43 GiB, 2 GiB, and 39 GiB, respectively. Partition **/dev/vdb1**'s total capacity, used capacity, and available capacity are 43 GiB, 51 MiB, and 40 GiB, respectively.

Viewing Disk Usage in Windows Server 2008

In this section, Windows Server 2008 R2 Enterprise 64bit is used as an example.

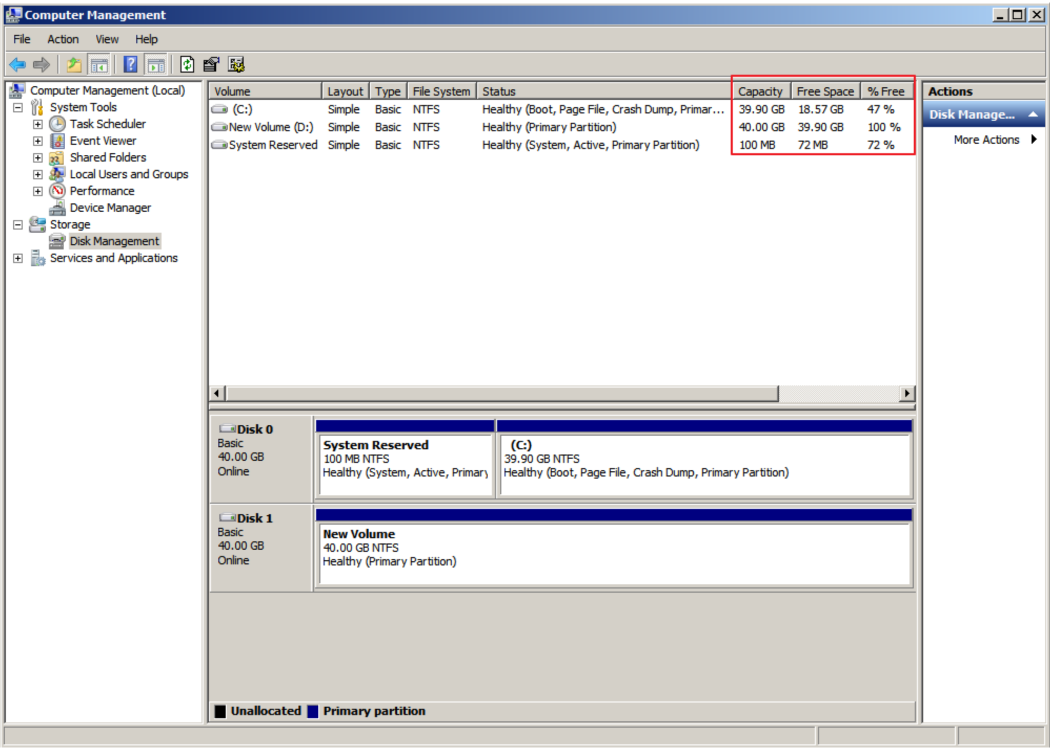
Step 1 On the desktop of the server, right-click **Computer** and choose **Manage** from the shortcut menu.

The **Server Manager** window is displayed.

Step 2 In the navigation tree, choose **Storage** > **Disk Management**.

The sizes and available spaces of the volumes on the current disk are displayed in the middle pane.

Figure 19-7 Disk Management page



----End

Viewing Disk Usage in Windows Server 2016

In this section, Windows Server 2016 Standard 64bit is used as an example.

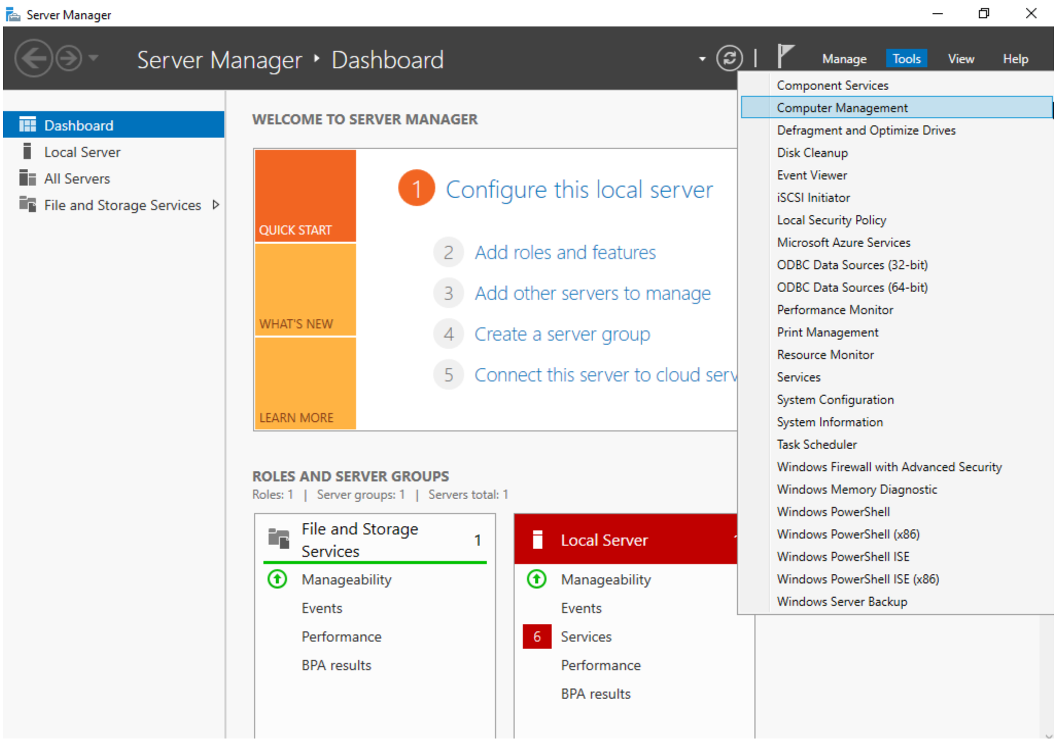
Step 1 On the desktop of the server, click the start icon in the lower left corner.

The **Windows Server** window is displayed.

Step 2 Click **Server Manager**.

The **Server Manager** window is displayed.

Figure 19-8 Server Manager page

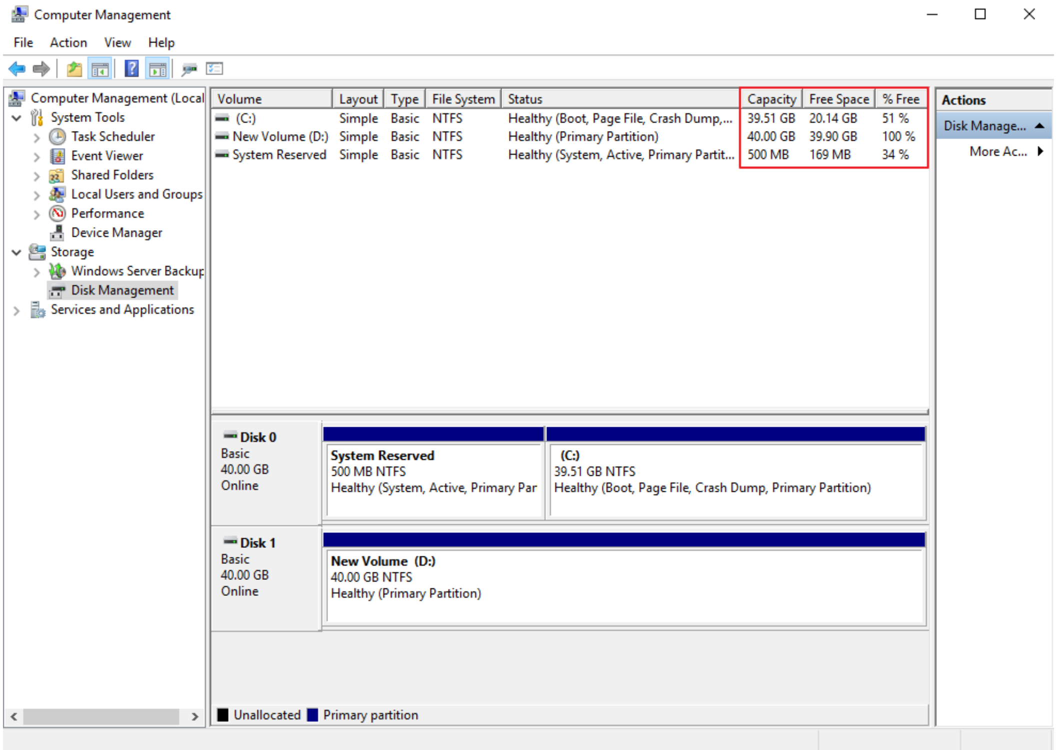


Step 3 In the upper right corner, choose **Tools** > **Computer Management**.

Step 4 Choose **Storage** > **Disk Management**.

In the middle pane, you can view the sizes and available spaces of the volumes on the disk.

Figure 19-9 Disk list page



----End

Installing Agent to View Disk Usage

Some disk monitoring metrics require that the agent to be installed.

For details about how to install the Agent on an ECS, see section "Installing and Configuring the Agent on a Linux ECS or BMS" in the *Cloud Eye User Guide*.

Table 19-9 Disk metrics

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_free	(Agent) Available Disk Space	Free space on the disks Unit: GB • Linux: Run the **df -h** command to check the value in the **Avail** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). • Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_total	(Agent) Disk Storage Capacity	Total space on the disks, including used and free Unit: GB ● Linux: Run the **df -h** command to check the value in the **Size** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_used	(Agent) Used Disk Space	Used space on the disks Unit: GB ● Linux: Run the **df -h** command to check the value in the **Used** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_usedPercent	(Agent) Disk Usage	Percentage of total disk space that is used, which is calculated as follows: **Disk Usage** = **Used Disk Space**/**Disk Storage Capacity** Unit: percent ● Linux: It is calculated as follows: Used/Size. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	0-100	ECS - Mount point	1 minute

19.8.6 How Can I Monitor My Disk Usages?

Some disk monitoring metrics require that the agent to be installed.

For details about how to install the Agent on an ECS, see section "Installing and Configuring the Agent on a Linux ECS or BMS" in the *Cloud Eye User Guide*.

Table 19-10 Disk metrics

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_free	(Agent) Available Disk Space	Free space on the disks Unit: GB • Linux: Run the **df -h** command to check the value in the **Avail** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). • Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_total	(Agent) Disk Storage Capacity	Total space on the disks, including used and free Unit: GB ● Linux: Run the **df -h** command to check the value in the **Size** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_used	(Agent) Used Disk Space	Used space on the disks Unit: GB ● Linux: Run the **df -h** command to check the value in the **Used** column. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	≥ 0	ECS - Mount point	1 minute

Metric	Parameter	Description	Value Range	Monitored Object & Dimension	Monitoring Period (Raw Data)
disk_usedPercent	(Agent) Disk Usage	Percentage of total disk space that is used, which is calculated as follows: **Disk Usage** = **Used Disk Space**/**Disk Storage Capacity** Unit: percent ● Linux: It is calculated as follows: Used/Size. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~). ● Windows: Use the WMI interface to call GetDiskFreeSpaceExW API to obtain disk space data. The path of the mount point prefix cannot exceed 64 characters. It must start with a letter, and contain only digits, letters, hyphens (-), dots (.), and swung dashes (~).	0-100	ECS - Mount point	1 minute

19.8.7 Can I Transfer the Data Disk Capacity to a System Disk?

Sorry, you cannot.

Currently, the capacity of an EVS disk cannot be transferred to another disk. Multiple EVS disks cannot be combined into a single, larger disk, either.

Common Scenarios

1. You want to expand the system disk capacity, but created a new data disk.
2. You want to expand the system disk capacity, but expanded the data disk capacity.

Recommended Solution

- If you do not need the data on the data disk, you can just delete the data disk, and then **expand the system disk**.
- If you need the data on the data disk, you can create a small-capacity data disk, copy data from the original data disk to the new data disk, and then expand the system disk capacity.
  a. Back up the data disk using the CBR service.
  b. Create a new data disk with the desired capacity and attach it to the server. After initializing the disk, copy the data from the original data disk to the new data disk.
  c. Confirm that the services on the new data disk are available. Then, delete the original data disk, and delete the created backup.
  d. Expand the system disk capacity by referring to **Disk Capacity Expansion**.

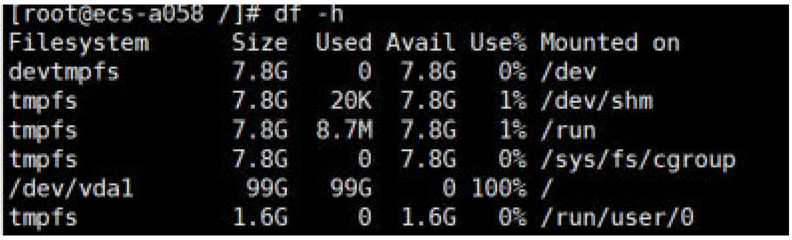
19.8.8 Why the Space of My New Disk Is Full After I Uploaded Only 500 MB of Files to the Disk?

Troubleshoot this issue by performing the following steps:

1. Check whether the disk partition usage is 100% or almost 100%.

```
df -h
```

Figure 19-10 Checking the partition usage



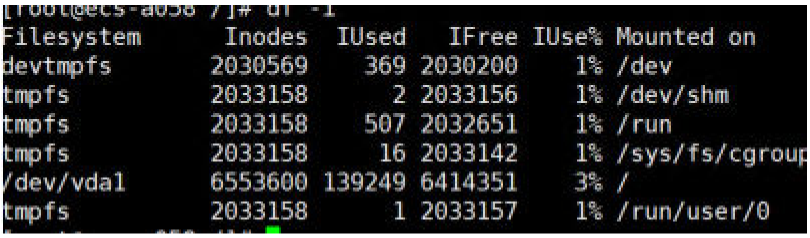
```
[root@ecs-a058 /]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        7.8G     0  7.8G   0% /dev
tmpfs           7.8G   20K  7.8G   1% /dev/shm
tmpfs           7.8G  8.7M  7.8G   1% /run
tmpfs           7.8G     0  7.8G   0% /sys/fs/cgroup
/dev/vda1        99G   99G     0 100% /
tmpfs           1.6G     0  1.6G   0% /run/user/0
```

In this example, the **/dev/vda1** partition usage is 100%.

2. Check the disk space usage.

```
df -i
```

Figure 19-11 Checking the disk space usage



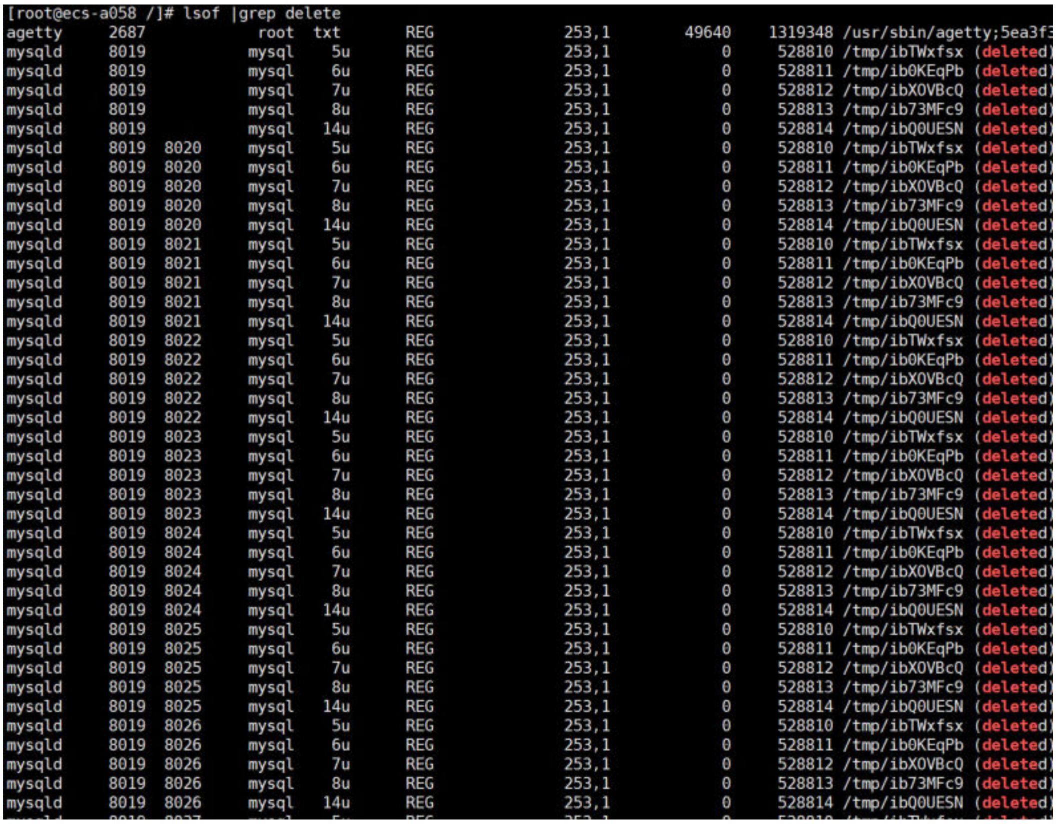
```
Filesystem      Inodes  IUsed   IFree IUse% Mounted on
devtmpfs       2030569    369 2030200    1% /dev
tmpfs          2033158      2 2033156    1% /dev/shm
tmpfs          2033158    507 2032651    1% /run
tmpfs          2033158     16 2033142    1% /sys/fs/cgroup
/dev/vda1      6553600 139249 6414351    3% /
tmpfs          2033158      1 2033157    1% /run/user/0
```

In this example, the disk space usage is low.

3. Check the deleted process files in the system.

```
lsof | grep deleted
```

Figure 19-12 Checking the deleted process files in the system



```
[root@ecs-a058 /]# lsof |grep delete
agetty     2687             root  txt    REG    253,1    49640    1319348 /usr/sbin/agetty;5ea3f
mysqld     8019            mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019            mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019            mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019            mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019            mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8020      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8020      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8020      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8020      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8020      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8021      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8021      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8021      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8021      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8021      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8022      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8022      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8022      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8022      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8022      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8023      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8023      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8023      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8023      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8023      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8024      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8024      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8024      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8024      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8024      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8025      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8025      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8025      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8025      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8025      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
mysqld     8019  8026      mysql   5u    REG    253,1        0     528810 /tmp/ibTWxfsx (deleted)
mysqld     8019  8026      mysql   6u    REG    253,1        0     528811 /tmp/ib0KEqPb (deleted)
mysqld     8019  8026      mysql   7u    REG    253,1        0     528812 /tmp/ibXOVBcQ (deleted)
mysqld     8019  8026      mysql   8u    REG    253,1        0     528813 /tmp/ib73MFc9 (deleted)
mysqld     8019  8026      mysql  14u    REG    253,1        0     528814 /tmp/ibQOUESN (deleted)
```

Roughly calculate the total size of the deleted files based on the returned command output. If it is almost the same as the used space of the disk, the disk space may be used up by the deleted processes that have not been released.

4. Go to the location of a deleted file to check whether the file is still there.

   ll */tmp/*

   Note that variable */tmp/* in the command indicates the path of the deleted file.

5. If the file is not there, run the following command to terminate the process, or restart the server to release the used space.

   kill -9 *PID*

   Note that variable *PID* in the command indicates the process ID.

6. Check that the process is terminated.

   lsof | grep deleted

7. Check that the disk partition usage is no longer 100%.

   df -h

19.9 Performance

19.9.1 How Can I Test My Disk Performance?

Precautions

In the disk performance test, if the start sector number is not 4-KiB aligned, the disk performance will be greatly affected. Ensure that the start sector number is 4-KiB aligned before you start the test.

NOTE

To test the performance of a shared disk, the following requirements must be met:

- The shared disk must be attached to multiple servers (ECSs or BMSs).
- If the shared disk is attached to multiple ECSs, these ECSs must belong to the same anti-affinity ECS group.

  If these ECSs fail to meet the anti-affinity requirement, the shared disk cannot reach the optimal performance.

The testing process for Windows and Linux is different.

- **Windows**
- **Linux**

Windows

The way you test disk performance depends on the server OS. This section uses Windows Server 2019 Standard 64-bit as an example. For other Windows OSs, see the corresponding OS documentations.

Install the performance measurement tool Iometer before the test. You can obtain the tool at **http://www.iometer.org/**.

Step 1 Log in to the server.

Step 2 Press **win+R** to open the **Run** window. Enter **msinfo32** and click **OK**.

The system information window is displayed.

Step 3 Choose **components** > **storage** > **disks**. In the right pane, view the partition offset.

- If 4096 can be divided by the parameter value, the partition is 4-KiB aligned. Go to **Step 4**.
- If 4096 cannot be divided by the parameter value, the partition is not 4-KiB aligned. Ensure 4-KiB alignment for the partition before continuing the test.

NOTICE

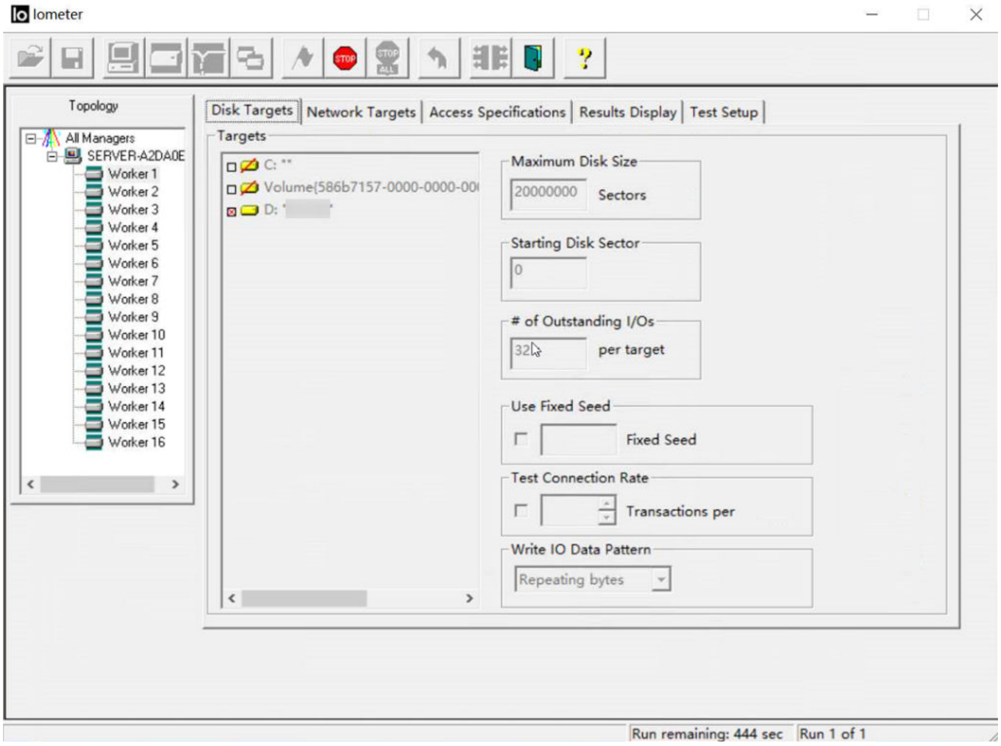
If you delete the partition and select another start sector number for 4-KiB alignment, you will lose all the data on that partition. Exercise caution when performing this operation.

Step 4 Use Iometer to test the disk performance. For details, see the Iometer product documentation.

When the disk IOPS and throughput are tested, the parameter settings for Iometer and fio are the same. For details, see **Table 19-11**.

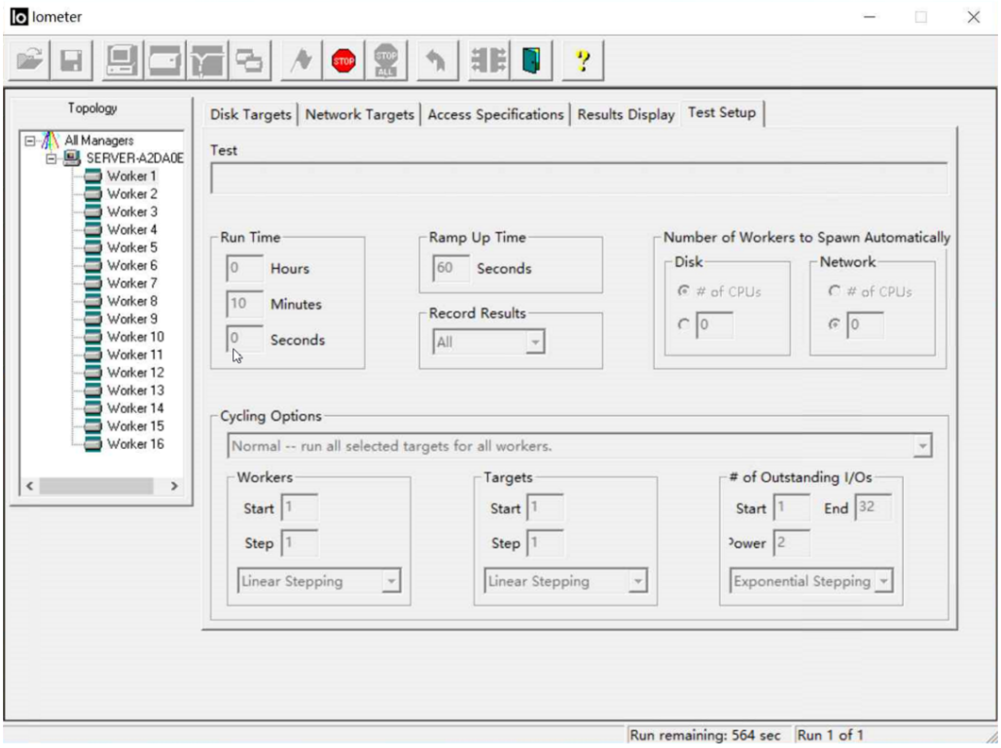
The following example uses Iometer to test the disk performance.

1. Set the workflow.

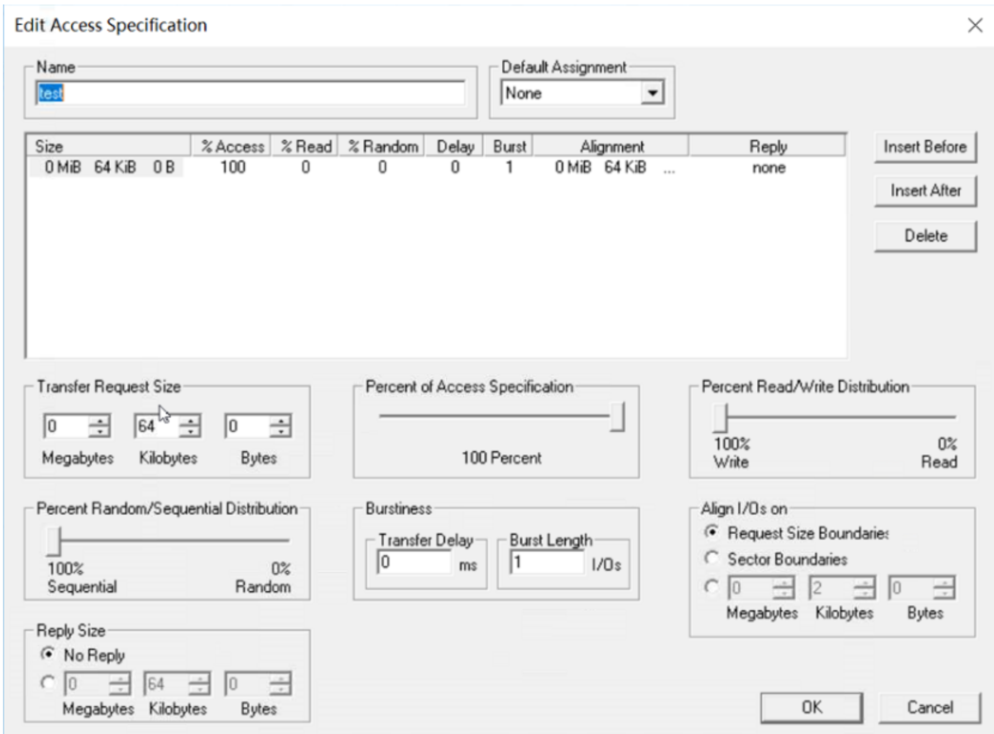


2. Set the test run time.

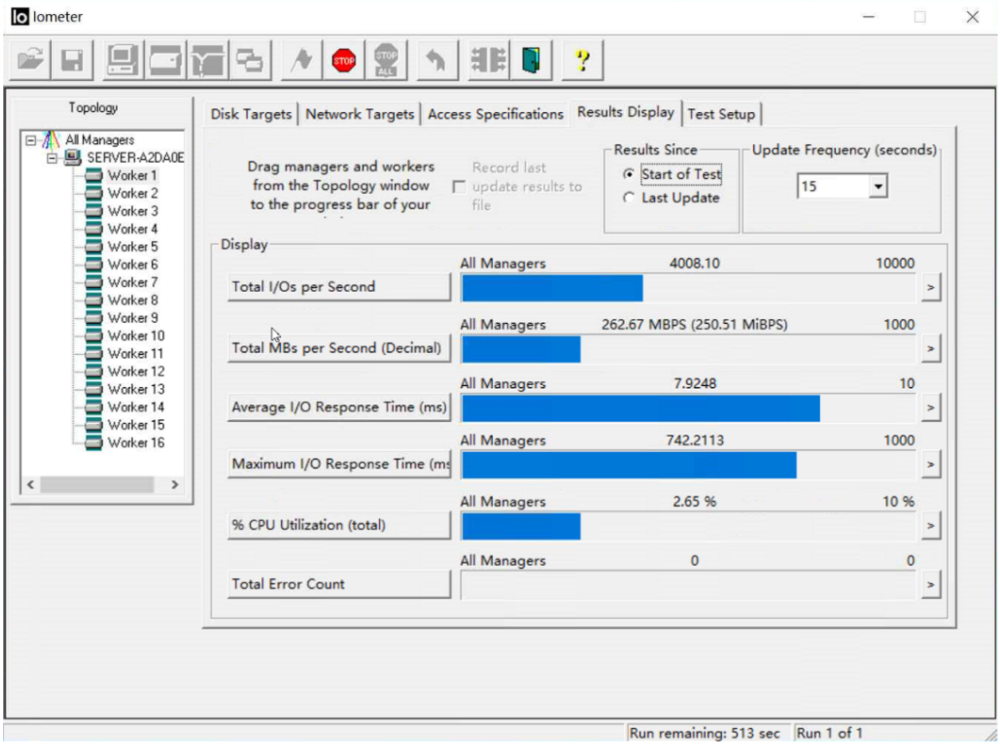
   In this example, the test run time is set to 10 minutes, with 60 seconds ramp up time. Disk performance is tested after the writes are stable.



3. Set the data block size and read/write policy. In this example, the disk size is set to 64 KiB, the policy is 100% sequential write.



4. View the test results.



----End

Linux

If you use an old version Linux OS, for example CentOS 6.5, and run **fdisk** to create partitions, the default start sector number will not be 4-KiB aligned, which will greatly affect the test performance. For this reason, if such an OS is used, you are advised to select a new start sector number, one that is 4-KiB aligned, when creating partitions.

The way you test disk performance depends on the server OS. This section uses CentOS 7.2 64-bit as an example. For other Linux OSs, see the corresponding OS documentations.

Install the performance measurement tool, fio, before the test.

Step 1 Log in to the server and switch to user **root**.

Step 2 Before you start the test, run the following command to check whether the start sector number is 4-KiB aligned:

fdisk -lu

Information similar to the following is displayed:

```
[root@ecs-centos sdc]# fdisk -lu

Disk /dev/xvda: 10.7 GiB, 10737418240 bytes, 20971520 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: 0x7db77aa5

   Device Boot      Start         End      Blocks   Id  System
/dev/xvda1   *        2048    20968919    10483436   83  Linux
```

```
Disk /dev/xvdb: 10.7 GiB, 10737418240 bytes, 20971520 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 /dev/xvdc: 53.7 GiB, 53687091200 bytes, 104857600 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: 0x3cf3265c

  Device Boot      Start         End      Blocks   Id  System
/dev/xvdc1          2048    41943039    20970496   83  Linux
```

- If 8 can be divided by the start sector number, the number is 4-KiB aligned. Go to **Step 3**.
- If 8 cannot be divided by the start sector number, the number is not 4-KiB aligned. Delete the partition and select a 4-KiB aligned start sector number for the new partition before continuing the test.

NOTICE

If you delete the partition and select another start sector number for 4-KiB alignment, you will lose all the data on that partition. Exercise caution when performing this operation.

Step 3 Run the following commands and use fio to test the disk performance:

- To test random write IOPS, run the following command: **fio -direct=***1* **-iodepth=***128* **-rw=***randwrite* **-ioengine=***libaio* **-bs=***4k* **-size=***10G* **-numjobs=***1* **-runtime=***600* **-group_reporting -filename=***/opt/fiotest/fiotest.txt* **-name=***Rand_Write_IOPS_Test*
- To test random read IOPS, run the following command: **fio -direct=***1* **-iodepth=***128* **-rw=***randread* **-ioengine=***libaio* **-bs=***4k* **-size=***10G* **-numjobs=***1* **-runtime=***600* **-group_reporting -filename=***/opt/fiotest/fiotest.txt* **-name=***Rand_Read_IOPS_Test*
- To test write throughput, run the following command: **fio -direct=***1* **-iodepth=***32* **-rw=***write* **-ioengine=***libaio* **-bs=***1024k* **-size=***10G* **-numjobs=***1* **-runtime=***600* **-group_reporting -filename=***/opt/fiotest/fiotest.txt* **-name=***Write_BandWidth_Test*
- To test read throughput, run the following command: **fio -direct=***1* **-iodepth=***32* **-rw=***read* **-ioengine=***libaio* **-bs=***1024k* **-size=***10G* **-numjobs=***1* **-runtime=***600* **-group_reporting -filename=***/opt/fiotest/fiotest.txt* **-name=***Read_BandWidth_Test*
- To test random read latency, run the following command: **fio -direct=***1* **-iodepth=***1* **-rw=***randread* **-ioengine***=libaio* **-bs=***4k* **-size=***10G* **-numjobs=***1* **-runtime=***60* **-group_reporting -filename=***/opt/fiotest/fiotest.txt* **-name=***Rand_Read_LATE_Test*

NOTICE

- When using fio to perform a raw disk performance test, ensure that no partitions and file systems have been created on the disk and there is no data stored on the disk. Or, the raw disk test will damage the file system, and data on the disk will become read-only. In this case, your only option will be to delete the disk and buy a new one to continue the test.
- Do not perform the test on a disk with service data on it. If such test is a must, you are advised to perform the test as follows:
  - Back up the disk data before the test as you may damage the data on the disk.
  - Specify a file, for example **-filename=/opt/fiotest/fiotest.txt**, to test the performance of the file system.

Table 19-11 lists the fio test parameters.

Table 19-11 Parameter description

Parameter	Description
direct	Defines whether direct I/O is used. – Set to **0**: buffered I/O is used. – Set to **1**: direct I/O is used.
iodepth	Defines the I/O queue depth. This queue depth refers to the queue depth of each thread regardless of whether a single or multiple threads are used in the test. Total concurrent I/Os of fio = iodepth x numjobs Examples: – If there is a single thread and **-iodepth=32**, the I/O queue depth of this thread is 32 and the total concurrent I/Os of fio is 32 (32 x 1). – If there are three threads and **-iodepth=32**, the I/O queue depth of each thread is 32 and the total concurrent I/Os of fio is 96 (32 x 3).
rw	Defines the test read/write policy. – **randread**: random read – **randwrite**: random write – **read**: sequential read – **write**: sequential write – **randrw**: mixed random read/write

Parameter	Description
ioengine	Defines how fio delivers the I/O request (synchronously or asynchronously). – Synchronous I/O: Only one I/O request is delivered at a time, and the response is returned after the kernel has processed the request. That said, the single-thread I/O queue depth is always less than 1, and multi-thread concurrent processing can be used to handle such issues. Normally, 16 to 32 concurrent working threads fully occupy the I/O queue depth. – Asynchronous I/O: Multiple I/O requests are delivered using libaio at a time. Wait for the process to complete and reduce the interaction times to improve efficiency.
bs	Defines the I/O block size. The unit can be KiB, Kb, MiB, and Mb, and the default value is 4 KiB.
size	Defines the amount of data processed by the test I/Os. If parameters, such as **runtime**, are not specified, the test ends until fio has processed all the specified data amount. The value can be a number with a unit or percentage. A number with a unit indicates the read/write data amount, for example **size=10G**, indicating a 10-GiB read/write data amount. A percentage indicates the ratio of read/write data amount to the total size of files, for example **size=20%**, indicating the read/write data amount takes 20% of the total file space.
numjobs	Defines the number of concurrent threads.
runtime	Defines the test time. If this parameter is not specified, the test ends until the specified data amount is processed by the block size defined using parameter **size**.
group_reporting	Defines the test result display mode. The parameter value displays the statistics on a single thread instead of that on all jobs.
filename	Defines the name of the test file or device. – If a file is specified, the performance of the file system is tested. Example: **-filename=/opt/fiotest/fiotest.txt** – If a device name is specified, the performance of the raw disk is tested. Example: **-filename=/dev/vdb** **NOTICE** If the test is performed on a disk already has partitions and file systems created as well as data on it, user parameter **filename** to specify a file so that the original file system is not damaged and the data is not overwritten.
name	Defines the test task name.

----End

19.9.2 Why Does My Disk Performance Test Using Fio Have Incorrect Results?

Symptom

You have followed the test performance method, but the test results do not meet expectations.

Troubleshooting

During a disk performance test, the disk and stress test conditions play an important role.

Possible causes are listed here in order of their probability.

If the fault persists after you have ruled out one cause, move on to the next one in the list.

NOTICE

Some operations may result in data loss. It is recommended that you use raw disks for performance test.

Figure 19-13 Troubleshooting

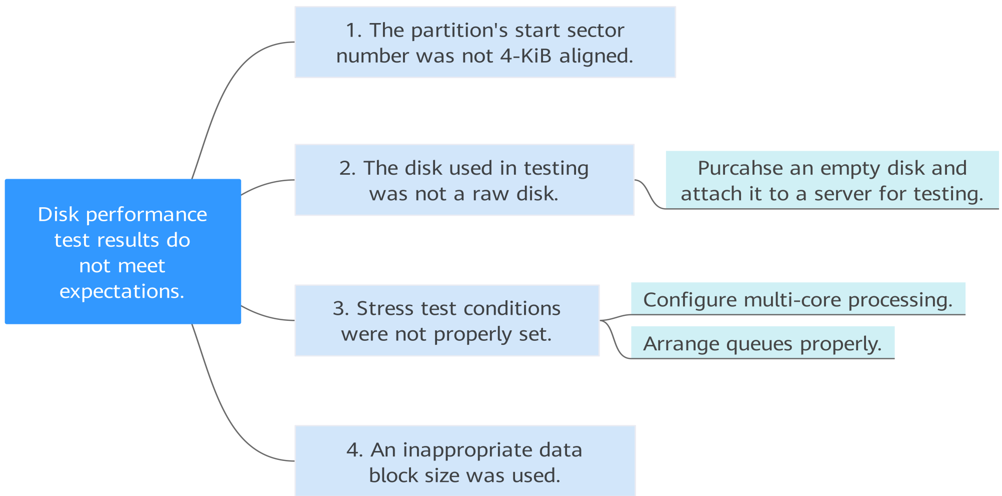


Table 19-12 Troubleshooting

Possible Cause	Solution
The partition's start sector number is not 4-KiB aligned.	Go to **Check Whether Partition's Start Sector Number Is 4-KiB Aligned**. Delete the partition and select a 4-KiB aligned start sector number for the new partition.
The disk used in testing was not a raw disk.	Purchase an empty disk and attach it to a server for testing.
Stress test conditions were not properly set.	Configure multi-core processing and arrange queues properly to maximize the concurrent performance.
An inappropriate data block size was used.	Set a suitable data block size. ● When testing the disk IOPS, set the data block size to a small value, for example, 4 KiB. ● When testing the disk throughput, set the data block size to a large value, for example, 1024 KiB.

Check Whether Partition's Start Sector Number Is 4-KiB Aligned

Step 1 Log in to the server and switch to user **root**.

Step 2 Before you start the test, run the following command to check whether the start sector number is 4-KiB aligned:

fdisk -lu

Information similar to the following is displayed:

```
[root@ecs-centos sdc]# fdisk -lu

Disk /dev/xvda: 10.7 GiB, 10737418240 bytes, 20971520 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: 0x7db77aa5

   Device Boot      Start         End      Blocks   Id  System
/dev/xvda1   *        2048    20968919    10483436   83  Linux

Disk /dev/xvdb: 10.7 GiB, 10737418240 bytes, 20971520 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 /dev/xvdc: 53.7 GiB, 53687091200 bytes, 104857600 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: 0x3cf3265c

   Device Boot      Start         End      Blocks   Id  System
/dev/xvdc1            2048    41943039    20970496   83  Linux
```


- If 8 can be divided by the start sector number, the number is 4-KiB aligned.
- If 8 cannot be divided by the start sector number, the number is not 4-KiB aligned. Delete the partition and select a 4-KiB aligned start sector number for the new partition before continuing the test.

NOTICE

If you delete the partition and select another start sector number for 4-KiB alignment, you will lose all the data on that partition. Exercise caution when performing this operation.

----End

19.9.3 How Can I Handle a Slowdown in Disk Read/Write Speed and Increased I/O?

Symptom

If you are aware of a service slowdown, depending on if you are examining a Windows or Linux server, you can take the following actions:

- Windows: Open **Task Manager** and view the average response time.
- Linux: Run **iostat -dx** to view the I/O performance.

If the disk read/write speed is slowed down, disk I/O increases, or the await time increases, the disk is likely encounters a performance bottleneck.

Solution

It is recommended that you change to a disk type with a higher specification.

If your disk contains important data, create a new disk from the disk backup, so you do not lose any of the disk data. The procedure is as follows:

1. Create a backup for the disk.
2. Create a new disk from this backup. During the creation, select a new disk type and configure advanced settings (sharing, SCSI, and encryption) based on your service requirements.

19.9.4 How Can I Improve My Disk Performance?

You can use the following methods to improve your disk performance:

See sections "RAID Array Creation with EVS Disks" and "Using LVM to Manage EVS Disks" in the *Elastic Volume Service Best Practices*.

19.9.5 Why My Disk's Read IOPS Can't Reach the Theoretical Maximum IOPS When the Disk I/O Usage Is Almost 100%?

Symptom

A 500 GiB ultra-high I/O disk had an I/O usage of 99.94%, but it only had 12,000 IOPS.

Description

- **100% disk I/O usage does not mean that the disk IOPS reaches the maximum.**

  Disk I/O usage calculates the read or write operations performed by a disk in a measurement period. It describes how busy a disk is, not the disk I/O performance.

  EVS disks can process I/O requests concurrently, so 100% disk I/O usage does not mean that the disk encounters the performance bottleneck. For example, an EVS disk takes 0.1 second to process an I/O request and can process 10 I/O requests concurrently. If 10 I/O requests are submitted serially, the disk takes 1 second to process all I/O requests. In this 1-second measurement period, the disk I/O usage reaches 100%. However, if 10 I/O requests are submitted concurrently, the disk takes just 0.1 second to process all the requests. This way, the disk I/O usage in a 1-second measurement period is only 10%. This means that a disk can still process I/O requests even if its I/O usage reaches 100%.
- **Why does the disk not reach the theoretical maximum IOPS?**

  The actual maximum IOPS that a disk can reach is calculated as follows: Disk IOPS = Min. (Max. IOPS, Min. IOPS + IOPS per GiB x Disk size). For a 500 GiB disk, its IOPS is calculated as follows: Disk IOPS = Min. (20,000, 1,500 + 50 × 500) = 20,000

  The disk read IOPS is the number of read operations performed by the disk per second. IOPS is also affected by latency. In a single-queue access scenario with 4 KiB data blocks, the access latency of an ultra-high I/O disk is 1 ms, which means the disk can process 1,000 requests (IOPS) in a second. 12,000 IOPS indicates that the queue depth is 12. To reach the theoretical maximum IOPS (20,000), the queue depth should reach 20.

19.10 Sharing

19.10.1 Do I Have to Deploy a Cluster to Use Shared Disks?

Yes.

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.

Shared EVS disks do not have cluster management capabilities. You need to build a clustered system for data sharing, such as Windows MSCS, Veritas VCS, and Veritas CFS clusters.

19.10.2 How Many Servers Can I Attach a Shared Disk to?

A shared disk can be attached to up to 16 servers.

19.10.3 How Can I Attach a Shared Disk to Multiple Servers?

A shared disk can be attached to multiple servers on the management console. You can choose to attach it to servers one by one or in a batch.

For details, see **Attaching a Shared Disk**.

19.10.4 Can a Shared Disk Be Attached to Servers That Belong to Different Accounts?

No. A shared disk can only be attached to servers that belong to the same account and are in the same AZ.

19.10.5 Can I Attach a Shared Disk to Servers Running Different OSs?

- It is recommended that you do not simultaneously attach a shared disk to servers running Linux and Windows.
- If a shared disk is attached to servers running different versions of the same OS type, it can be used normally. For example, a shared disk attached to one server running CentOS 6 and another server running CentOS 7 can work fine.

19.11 Backup

19.11.1 Do I Need to Stop the Server Before Performing a Backup?

No. You can back up servers that are in use. When a server is running, data is written into disks on the server, and some newly generated data is cached in the server memory. During a backup task, data in the memory will not be automatically written into disks, so the disk data and their backups may be inconsistent.

To ensure data integrity, you are advised to perform the backup during off-peak hours when no data is written to the disks. For applications that require strict consistency, such as databases and email systems, you are advised to enable application-consistent backup.

19.11.2 Can I Back Up and Restore My EVS Disk to a Different Region?

EVS disks already attached to ECSs support cross-region backup and restoration. Such disks can be backed up together with their ECSs using CBR cloud server backup, which supports cross-region replication. You can replicate server backups to your desired region and use the replicated backups to create images and provision servers.

EVS disks that have not been attached to ECSs do not support cross-region backup and restoration.

19.11.3 How Do I View My Backup Data?

You can check your backup data in the following ways:

NOTE

Backup data cannot be viewed on the CBR console.

Server Backups

1. Create an image from a server backup. For details, see section "Using a Backup to Create an Image" in the *Cloud Backup and Recovery User Guide*.
2. Use the image to create a server. For details, see section "Creating an ECS from an Image" in the *Image Management Service User Guide*.
3. Log in to the server to view the data.

Disk Backups

1. Create a new disk from a disk backup. For details, see section "Using a Backup to Create a Disk" in the *Cloud Backup and Recovery User Guide*.
2. Attach the disk to a server. For details, see section "Attaching a Non-Shared Disk" or "Attaching a Shared Disk" in the *Elastic Volume Service Getting Started*.
3. Log in to the server to view the data.

SFS Turbo Backups

1. Create a new SFS Turbo file system from an SFS turbo backup. For details, see section "Using a Backup to Create a File System" in the *Cloud Backup and Recovery User Guide*.
2. Mount the file system to a server.
   - To mount the file system to a Linux server, see section "Mounting an NFS File System to ECSs (Linux)" in the *Scalable File Service Getting Started*.
   - To mount the file system to a Windows server, see section "Mounting an NFS File System to ECSs (Windows)" in the *Scalable File Service Getting Started*.
3. Log in to the server to view the data.

A Appendix

A.1 EVS Disk Status

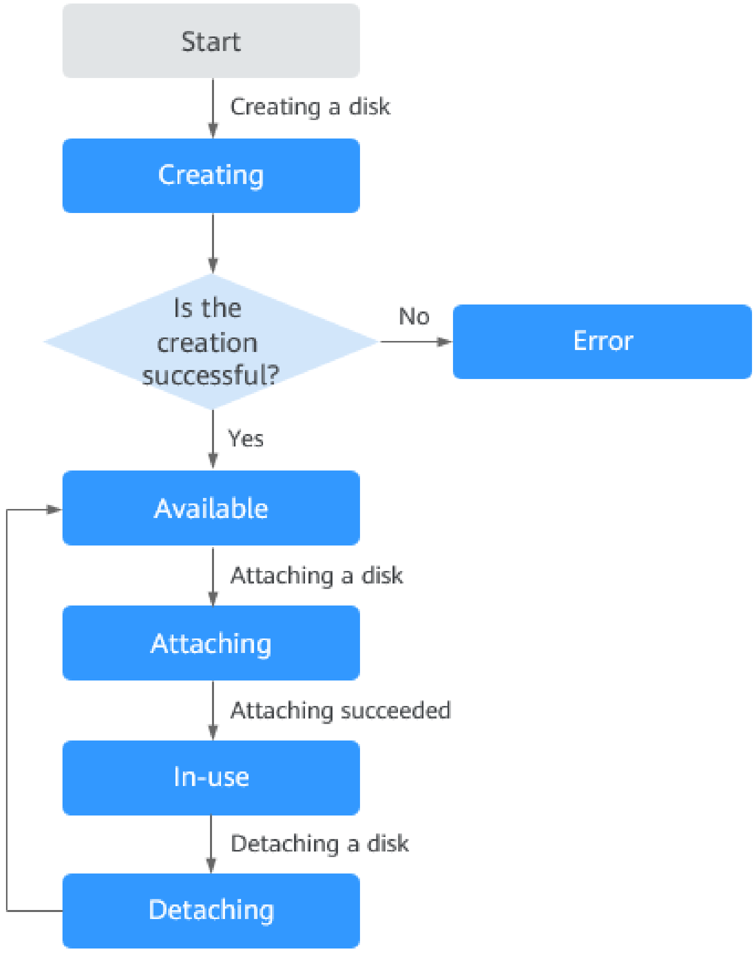
An EVS disk has several statuses. **Table A-1** lists EVS disk statuses, the meaning of each status, and the operations a disk in each status allows.

Table A-1 Disk status details

EVS Disk Status	Status Description	Allowed Operation
In-use	The EVS disk is attached to a server and is in use.	• Detaching • Creating backups • Expanding capacity **NOTE** A shared **In-use** EVS disk can be attached.
Available	The EVS disk has not been attached to any server and can be attached.	• Attaching • Expanding capacity • Deleting • Creating backups
Creating	The EVS disk is being created.	None
Attaching	The EVS disk is being attached to a server.	None
Detaching	The EVS disk is being detached from a server.	None
Deleting	The EVS disk is being deleted.	None
Restoring	A backup is being used to restore the EVS disk.	None

EVS Disk Status	Status Description	Allowed Operation
Expanding	The capacity of the EVS disk is being expanded.	None
Uploading	Data on the EVS disk is being uploaded to an image. This status occurs when you create an image from a server.	None
Downloading	Data is being downloaded from an image to the EVS disk. This status occurs when you create a server.	None
Error	An error occurs when you try to create an EVS disk.	Deleting
Deletion failed	An error occurs when you try to delete the EVS disk.	None
Expansion failed	An error occurs when you try to expand the capacity of the EVS disk.	Deleting
Restoration failed	An error occurs when you try to restore the EVS disk from a backup.	Deleting

Figure A-1 Change of some EVS disk statuses



NOTE

If an EVS disk status is **Error**, **Deletion failed**, **Expansion failed**, or **Rollback failed**, you can rectify the error by referring to **What Should I Do If an Error Occurs on My EVS Disk?**.

B Change History

Released On	Description
2023-07-30	This issue is the ninth official release, which incorporates the following change: Added the following content: Added the content about EVS billing in sections **Billing for Disks**, **Pay-per-Use Disk Usage Suggestions**, **How Can I Stop Being Billed for My Disk?**, **Will I Be Billed If I Have Purchased an EVS Disk But Not Used It?**, **Can I Recover My Disk Data If the Disk Is Deleted by Mistake?**, and **Will My EVS Disk Be Deleted When I Delete Its Server?**.
2023-03-13	This issue is the eighth official release, which incorporates the following changes: Updated the following content: ● Optimized some descriptions.
2018-09-30	This issue is the seventh official release, which incorporates the following changes: ● Added support for the EVS encryption function. ● Added support for expanding **In-use** EVS disks. ● Deleted the content of EVS replication as EVS replication APIs have been deprecated. ● Added support for the TMS tag function.
2018-03-30	This issue is the sixth official release, which incorporates the following change: ● Added the content of "Calculating the IOPS Limit of an EVS Disk".
2018-02-11	This issue is the fifth official release, which incorporates the following change: ● Added descriptions about EVS replication.

Released On	Description
2017-11-30	This issue is the fourth official release, which incorporates the following change: ● Added the operation guidance on initializing disks and performing post-expansion operations using the parted partitioning tool.
2017-08-30	This issue is the third official release, which incorporates the following changes: ● Added support for SCSI EVS disks. ● Added support for shared EVS disks.
2017-03-30	This issue is the second official release, which incorporates the following change: ● Added the **Rolling back** and **Rollback failed** states for EVS disks.
2017-01-06	This issue is the first official release.

