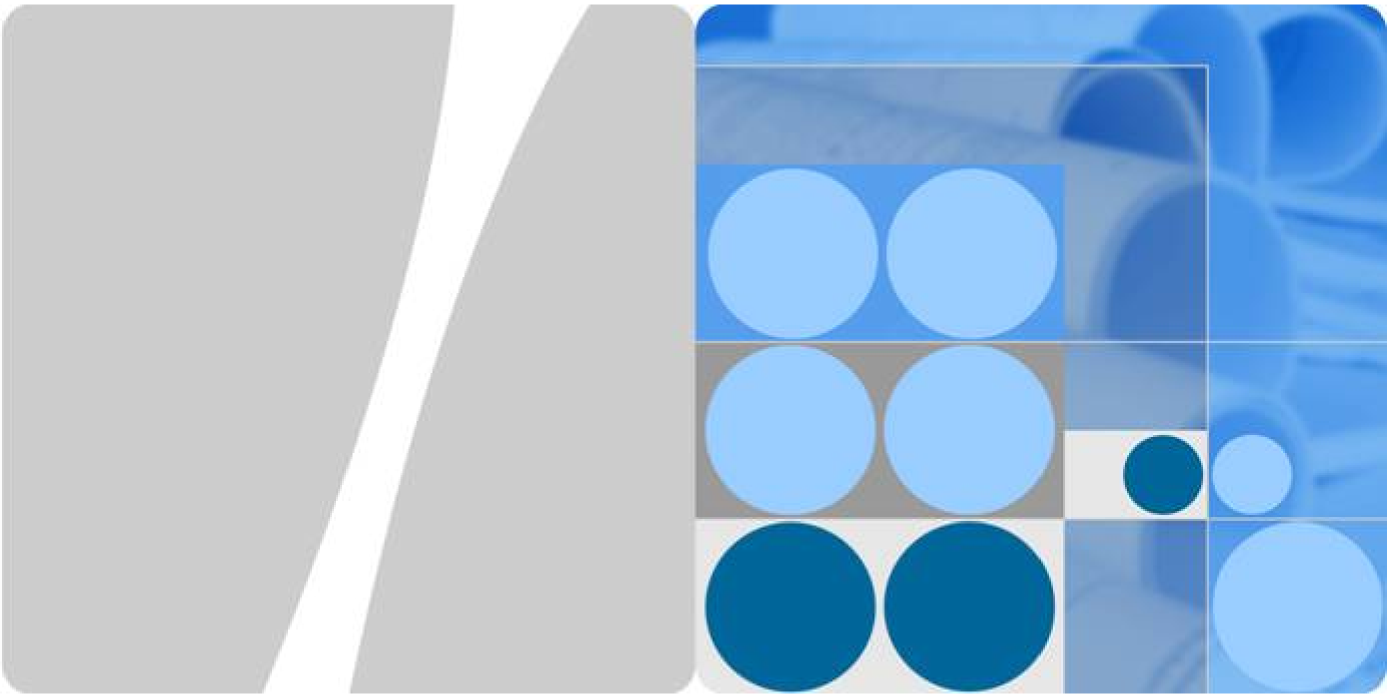


Scalable File Service

User Guide

Date **2023-09-20**

Contents

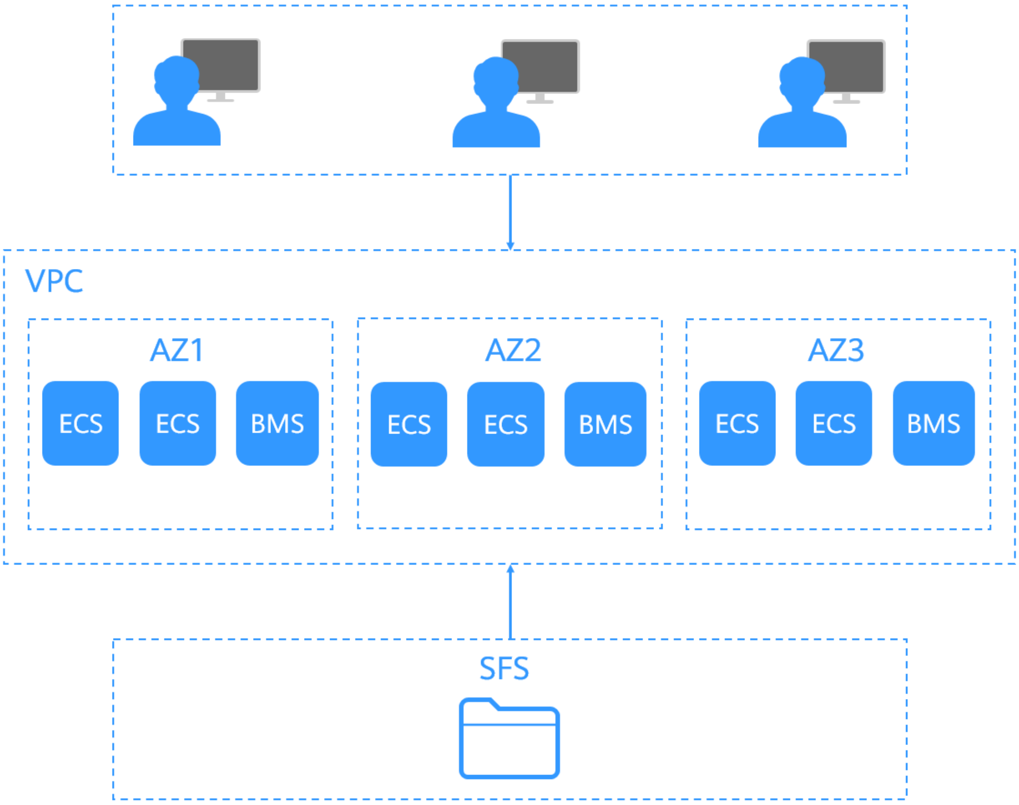
1 Introduction

1.1 What Is SFS?

Overview

Scalable File Service (SFS) provides scalable, high-performance (NAS) file storage. With SFS, you can enjoy shared file access spanning multiple Elastic Cloud Servers (ECSs), Bare Metal Servers (BMSs), and containers created on Cloud Container Engine (CCE). See **Figure 1-1**.

Figure 1-1 Accessing SFS



Compared with traditional file sharing storage, SFS has the following advantages:

- File sharing

  Servers in multiple availability zones (AZs) of a same region can access the same file system concurrently and share files.

- Elastic scaling

  Storage can be scaled up or down on demand to dynamically adapt to service changes without interrupting applications. You can complete resizing with a few clicks.

- Superior performance and reliability

  The service enables file system performance to increase as capacity grows, and delivers a high data durability to support rapid service growth.

- Seamless integration

  SFS supports NFS and CIFS protocols. With these standard protocols, a broad range of mainstream applications can read data from and write data to the file system. In addition, the service is compatible with SMB 2.0, SMB 2.1, and SMB 3.0, facilitating Windows clients to access the shared space.

- Easy operation

  In an intuitive graphical user interface (GUI), you can create and manage file systems with ease.

Accessing SFS

You can access SFS on the management console or via APIs by sending HTTPS requests.

- APIs

  Use APIs if you need to integrate SFS into a third-party system for secondary development. For detailed operations, see *Scalable File Service API Reference*.

- Management console

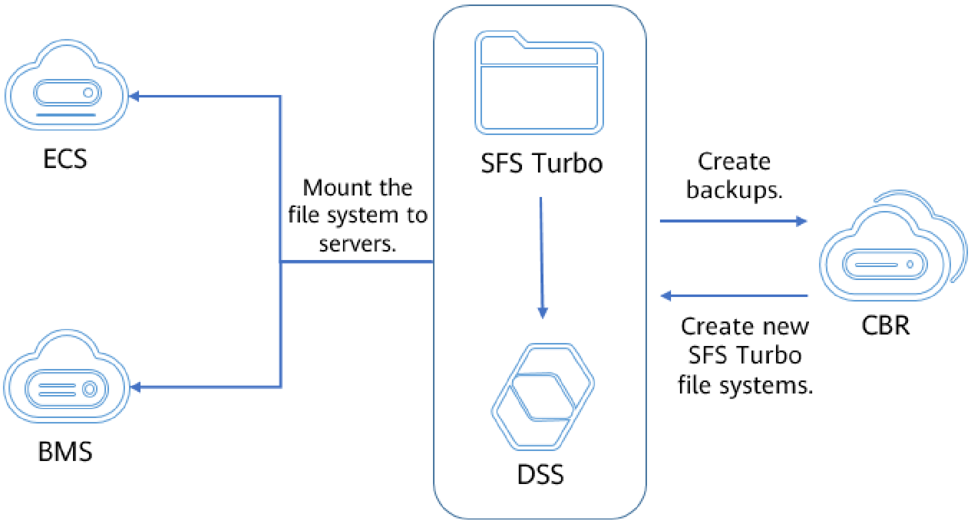
  Use the console if you prefer a web-based UI to perform operations.

1.2 Dedicated SFS Turbo

Overview

Dedicated SFS Turbo provides shared file storage for enterprises, governments, and finance institutions based on dedicated compute and storage resource pools. Dedicated resource pools are physically isolated from public pools. The reliable, efficient cloud experience dedicated pools offer can help you meet specific performance, application, and compliance needs.

Figure 1-2 Architecture of Dedicated SFS Turbo


ECS
BMS
Mount the file system to servers.
SFS Turbo
DSS
Create backups.
Create new SFS Turbo file systems.
CBR

Functions

- A variety of specifications

  Various file system types, including Standard, Performance, are available for diverse application workloads.

- Elastic scaling

  File system capacity can be increased on demand, and file system performance improved linearly.

- Reliable and secure

  Three-copy redundancy ensures 99.9999999% durability.

  Storage pool data encryption protects your data security.

  VPC isolation guarantees 100% isolation between tenants.

  Physically isolated storage pools provide exclusive resources for tenants.

- Backup and restore

  Dedicated SFS Turbo can be backed up using CBR. You can use backups to restore file systems.

- Monitoring

  Dedicated SFS Turbo can be interconnected with Cloud Eye, which allows you to view metrics including bandwidth, IOPS, and capacity.

- Auditing

  Dedicated SFS Turbo can be audited using CTS. You can view, audit, and backtrack file system operations.

Performance

Table 1-1 Performance

Specifications	Dependent Underlying Resources	Performance
Dedicated SFS Turbo Standard	DCC: C7, C7n, C6, C6s, and C3 instances DSS: high I/O storage pool	Bandwidth = Min. (1 GB/s, Available bandwidth of the DSS storage pool) IOPS = Min. (15,000, Available IOPS of the DSS storage pool)
Dedicated SFS Turbo Performance	DCC: C7, C7n, C6, C6s, and C3 instances DSS: ultra-high I/O storage pool	Bandwidth = Min. (2 GB/s, Available bandwidth of the DSS storage pool) IOPS = Min. (20,000, Available IOPS of the DSS storage pool)

NOTE

The available bandwidth and IOPS of a storage pool are in direct proportion to the storage capacity. When purchasing Dedicate SFS Turbo and planning DSS resources, reserve enough Dedicated SFS Turbo storage space and performance to prevent affecting the file system performance.

1.3 Application Scenarios

SFS Capacity-Oriented

Expandable to petabytes, SFS Capacity-Oriented provides fully hosted shared file storage. It features high availability and durability, and seamlessly handles data-intensive and bandwidth-intensive applications. It is suitable for multiple scenarios, including high-performance computing (HPC), media processing, file sharing, as well as content management and web services.

- HPC

  In industries that require HPC, such as simulation experiments, biopharmacy, gene sequencing, image processing, and weather forecast, SFS provides superb compute and storage capabilities, as well as high bandwidth and low latency.

- Media processing

  Services of TV stations and new media are more likely to be deployed on cloud platforms than before. Such services include streaming media, archiving, editing, transcoding, content distribution, and video on demand (VoD). In such scenarios, a large number of workstations are involved in the whole program production process. Different operating systems may be used by different workstations, requiring file systems to share materials. In addition, HD/4K videos have become a major trend in the broadcasting and TV industry. Taking video editing as an example, to improve audiences' audiovisual experience, HD editing is being transformed to 30- to 40-layer editing. A single editing client may require a file system with a bandwidth up to hundreds of MB per second. Usually, producing a single TV program needs several editing clients to process a lot of video materials concurrently. To meet such requirement, SFS provides customers with stable, bandwidth-intensive, and latency-sensitive performance.

- Content management and web service

  SFS can be used in various content management systems to store and provide information for websites, home directories, online releases, and archiving.

- Big data and analytic applications

  SFS delivers an aggregate bandwidth of up to 10 GB/s, capable of handling ultra-large data files such as satellite images. In addition, SFS has robust reliability to prevent service interruptions due to system failures.

SFS Turbo

Expandable to 320 TB, SFS Turbo provides a fully hosted shared file storage. It features high availability and durability to support massive small files and applications requiring low latency and high IOPS. SFS Turbo is perfect to scenarios such as high-performance websites, log storage, compression and decompression, DevOps, enterprise offices, and container applications.

- High-performance websites

  For I/O-intensive website services, SFS Turbo can provide shared website source code directories for multiple web servers, enabling low-latency and high-IOPS concurrent share access.

- Log storage

  SFS Turbo can provide multiple service nodes for shared log output directories, facilitating log collection and management of distributed applications.

- DevOps

  The development directory can be shared to multiple VMs or containers, simplifying the configuration process and improving R&D experience.

- Enterprise offices

  Office documents of enterprises or organizations can be saved in an SFS Turbo file system for high-performance shared access.

1.4 File System Types

SFS provides two types of file systems: SFS Capacity-Oriented and SFS Turbo. SFS Turbo is classified into SFS Turbo Standard, SFS Turbo Standard – Enhanced, SFS Turbo Performance, and SFS Turbo Performance – Enhanced.

The following table describes the features, advantages, and application scenarios of these file system types.

Table 1-2 Comparison of file system types

File System Type	Storage Class	Feature	Highlight	Application Scenario
SFS Capacity-Oriented	-	● Maximum bandwidth: 2 GB/s; maximum IOPS: 2,000 ● Latency: 3 to 20 ms; maximum capacity: 4 PB ● With optimized features, it is suitable for services that require large capacity and high bandwidth. **NOTE** ● Latency refers to the minimum latency under low workload conditions. It is unstable. ● Large files refer to files larger than 10 MB, and large I/Os refer to I/Os larger than 1 MB.	Large capacity, high bandwidth, and low cost	Cost-sensitive workloads which require large-capacity scalability, such as media processing, file sharing, HPC, and data backup. For workloads dealing with massive small files, SFS Turbo is recommended.
SFS Turbo	SFS Turbo Standard	● Maximum bandwidth: 150 MB/s; maximum IOPS: 5,000 ● Latency: 2 to 5 ms; maximum capacity: 32 TB ● It is suitable for services with massive small files and services that require low latency.	Low latency and tenant exclusive	Workloads dealing with massive small files, such as code storage, log storage, web services, and virtual desktop
	SFS Turbo Standard - Enhanced	● Maximum bandwidth: 1 GB/s; maximum IOPS: 15,000 ● Latency: 2 to 5 ms; maximum capacity: 320 TB ● Enhanced bandwidth, IOPS, and capacity	Low latency, high bandwidth, and tenant exclusive	Workloads dealing with massive small files and those requiring high bandwidth, such as code storage, file sharing, enterprise office automation (OA), and log storage.

File System Type	Storage Class	Feature	Highlight	Application Scenario
	SFS Turbo Performance	● Maximum bandwidth: 350 MB/s; maximum IOPS: 20,000 ● Latency: 1 to 2 ms; maximum capacity: 32 TB ● With optimized features, it is suitable for services with massive small files and services that require low latency and high IOPS.	Low latency, high IOPS, and tenant exclusive	Workloads dealing with massive small files, and random I/O-intensive and latency-sensitive services, such as high-performance websites, file sharing, and content management
	SFS Turbo Performance - Enhanced	● Maximum bandwidth: 2 GB/s; maximum IOPS: 100,000 ● Latency: 1 to 2 ms; maximum capacity: 320 TB ● Enhanced bandwidth, IOPS, and capacity	Low latency, high IOPS, high bandwidth, and tenant exclusive	Workloads dealing with massive small files, and latency-sensitive and bandwidth-demanding workloads, such as image rendering, AI training, and enterprise OA.

1.5 File System Encryption

SFS provides you with the encryption function. You can encrypt data on the new file systems if needed.

Keys for encrypting file systems are provided by Key Management Service (KMS), which is secure and convenient. You do not need to establish and maintain key management infrastructure. If you want to use your own key material, use the key import function on the KMS console to create a custom key whose key material is empty and import the key material to the custom key. For details, see section "Importing Key Materials" in *Data Encryption Workshop User Guide*.

To use the file system encryption function, you can directly select the encryption function when creating an SFS Turbo file system without authorization.

Encryption Key

Keys provided by KMS for encrypting SFS Capacity-Oriented file systems include a default key and custom keys.

- Default key: SFS automatically creates a default key and names it **sfs/default**.

The default key cannot be disabled and does not support scheduled deletion.

- Custom keys: Existing or newly created custom keys. For details, see Creating a Custom Key in the *Data Encryption Workshop User Guide*.

  If the custom key used by the encrypted file system is disabled or scheduled for deletion, the file system can only be used within a certain period of time (30s by default). Exercise caution in this case.

An SFS Turbo file system does not have a default key. You can use your existing key or create a new key. For details, see section "Creating a Custom Key" in the *Data Encryption Workshop User Guide*.

Who Has the Rights to Encrypt File Systems?

- The security administrator who has the "Security Administrator" permission can grant the KMS access rights for encryption.
- A common user who does not have the "Security Administrator" permission needs to contact the system administrator to obtain the "Security Administrator" permission.

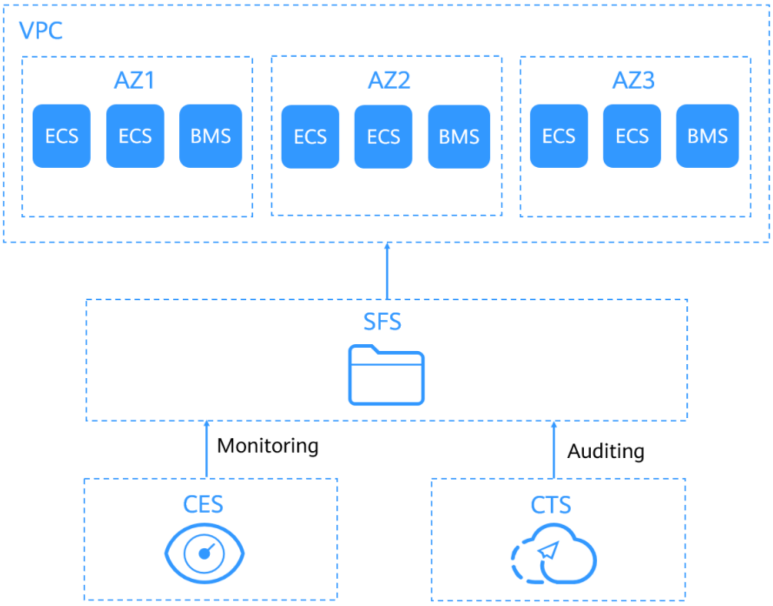
As long as the KMS access rights have been granted to SFS Capacity-Oriented, all common users in the same region can directly use the encryption function.

If there are multiple projects in the current region, the KMS access rights need to be granted to each project in this region.

1.6 SFS and Other Services

Figure 1-3 lists the relationship between SFS and other cloud services.

Figure 1-3 Relationships between SFS and other services



Relationships Between SFS and Other Services

Table 1-3 Related services

Function	Related Service	Reference
A file system and the servers must belong to the same project. File systems are mounted to shared paths for data sharing.	Elastic Cloud Server (ECS)	**Mounting an NFS File System to ECSs (Linux)** **Mounting an NFS File System to ECSs (Windows)** **Mounting a CIFS File System to ECSs (Windows)**
VPC provisions an isolated virtual network environment defined and managed by yourself, improving the security of cloud resources and simplifying network deployment. A server cannot access file systems in a different VPC. Before using SFS, assign the file system and the servers to the same VPC.	Virtual Private Cloud (VPC)	**Creating a File System**
IAM is an enterprise-level self-help cloud resource management system. It provides user identity management and access control functions. When an enterprise needs to provide SFS for multiple users within the enterprise, the enterprise administrator can use IAM to create users and control these users' permissions on enterprise resources.	Identity and Access Management (IAM)	**Permissions**
The encryption feature relies on KMS, which improves the data security of your file systems.	Data Encryption Workshop: Key Management Service (KMS)	**Encryption**
Once you have subscribed to SFS, you can monitor its performance, such as the read bandwidth, write bandwidth, and read write bandwidth on Cloud Eye, which does not require any plug-ins.	Cloud Eye	**Monitoring**

Function	Related Service	Reference
Cloud Trace Service (CTS) allows you to collect, store, and query cloud resource operation records and use these records for security analysis, compliance auditing, resource tracking, and fault locating. With CTS, you can record operations associated with SFS for later query, audit, and backtrack operations.	Cloud Trace Service (CTS)	**Auditing**

1.7 Basic Concepts

1.7.1 SFS Basic Concepts

Before you start, understand the following concepts.

NFS

Network File System (NFS) is a distributed file system protocol that allows different computers and operating systems to share data over a network.

CIFS

Common Internet File System (CIFS) is a protocol used for network file access. It is a public or open version of the Server Message Block (SMB) protocol, which is initiated by Microsoft. CIFS allows applications to access files on computers over the Internet and send requests for file services. Using the CIFS protocol, network files can be shared between hosts running Windows.

CIFS file systems cannot be mounted to Linux .

You are advised to use CIFS file systems in Windows OS.

File System

A file system provides users with shared file storage service through NFS and CIFS. It is used for accessing network files remotely. After a user creates a mount point on the management console, the file system can be mounted to multiple servers and is accessible through the standard POSIX.

POSIX

Portable Operating System Interface (POSIX) is a set of interrelated standards specified by Institute of Electrical and Electronics Engineers (IEEE) to define the application programming interface (API) for software compatible with variants of the UNIX operating system. POSIX is intended to achieve software portability at the source code level. That is, a program written for a POSIX compatible operating system may be compiled and executed on any other POSIX operating system.

DHCP

Dynamic Host Configuration Protocol (DHCP) is a LAN network protocol. The server controls an IP address range, and a client can automatically obtain the IP address and subnet mask allocated by the server when logging in to the server. By default, DHCP is not automatically installed as a service component of Windows Server. Manual installation and configuration are required.

1.7.2 Region and AZ

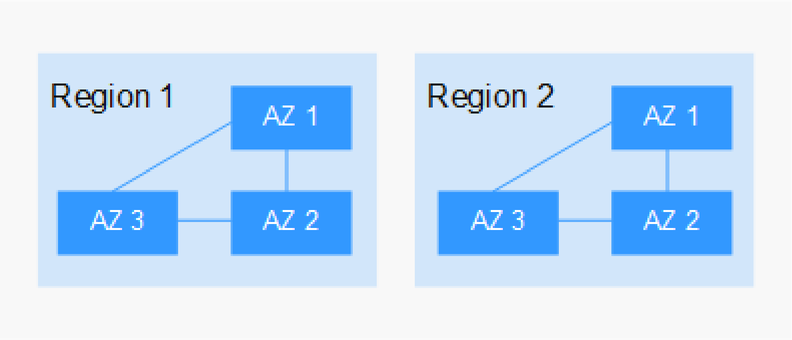
Concept

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 1-4 shows the relationship between regions and AZs.

Figure 1-4 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**.

1.8 Restrictions and Limitations

General

- SFS Capacity-Oriented supports the NFSv3 and CIFS protocols. Export options with NFSv3 include **rw**, **no_root_squash**, **no_all_squash**, and **sync**. Export options with CIFS include **rw** and **sync**.
- Encrypted CIFS file systems do not support copychunk.
- You can mount file systems to all ECSs that support the NFSv3 protocol. To obtain better performance, you are advised to use the operating systems listed in **Supported Operating Systems**, which have passed the compatibility test.
- CIFS file systems cannot be mounted to Linux .
- Currently, SFS does not support replication.
- Currently, SFS does not support cross-region access.
- SFS Capacity-Oriented is not suitable for file storage scenarios requiring low latency and high IOPS, such as database services, website building, and code storage.

SFS Capacity-Oriented

- Only NFSv3 is supported (NFSv4 is not supported), and CIFS is supported (SMB 2.0, 2.1, and 3.0 are supported, but SMB 1.0 is not supported).
- A file system can use either the NFS or CIFS protocol. It cannot use both protocols.
- A maximum of 10,000 compute nodes can be mounted to and access a single file system at the same time.
- Multi-VPC access is supported. You can add a maximum of 20 VPCs for one file system and create a maximum of 400 ACL rules for all added VPCs.

SFS Turbo

- Only the NFSv3 protocol is supported (NFSv4 is not supported).
- A maximum of 500 compute nodes can be mounted to and access a single file system at the same time.
- The maximum capacity of a single file system is 320 TB, and the maximum capacity of a single file is 16 TB.
- Maximum number of files supported by a single file system = Capacity/16 KB. For example, the maximum number of files supported by a 500 GB file system is 32,768,000 (500 GB/16 KB = 500 x 1024 x 1024/16).
- By default, a single directory can contain a maximum of 2 million files.

NOTE

If you need to execute the **ls**, **du**, **cp**, **chmod**, or **chown** command on a directory, you are advised to place no more than 500,000 files or subdirectories in that directory. Otherwise, requests may take long times as the NFS protocol sends a large number of requests to traverse directory files and requests are queueing up.

- The maximum full path is 1,024 bytes, and the maximum file name length is 255 bytes.
- The maximum soft link length is 1,024 bytes.
- The maximum number of hard links is 255.
- The maximum directory depth is 100 layers.

1.9 Permissions

If you need to assign different permissions to employees in your enterprise to access your SFS resources on the cloud, Identity and Access Management (IAM) is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you secure access to your cloud resources.

With IAM, you can use your cloud account to create IAM users, and assign permissions to the users to control their access to specific resources. For example, some software developers in your enterprise need to use SFS resources but should not be allowed to delete the resources or perform any other high-risk operations. In this scenario, you can create IAM users for the software developers and grant them only the permissions required for using SFS resources.

If your cloud account does not require individual IAM users for permissions management, skip this section.

IAM can be used free of charge. You pay only for the resources in your account. For more information about IAM, see *Identity and Access Management User Guide*.

SFS Permissions

By default, new IAM users do not have permissions assigned. You need to add a user to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specified operations on cloud services based on the permissions.

SFS is a project-level service deployed and accessed in specific physical regions. To assign SFS permissions to a user group, specify the scope as region-specific projects and select projects 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 SFS, the users need to switch to a region where they have been authorized to use this service.

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 SFS, see section "Permissions Policies and Supported Actions" in the *Scalable File Service API Reference*.

Table 1-4 lists all the system-defined roles and policies supported by SFS.

Table 1-4 System permissions for SFS Capacity-Oriented

Role/Policy Name	Description	Type	Dependency
SFS FullAccess	Administrator permissions for SFS. Users granted these permissions can perform all operations on file systems.	System-defined policy	None
SFS ReadOnlyAccess	Read-only permissions. Users granted these permissions can only view file system data.	System-defined policy	None

Role/Policy Name	Description	Type	Dependency
SFS Administrator	Permissions include: ● Creating, deleting, querying, and modifying file systems ● Adding, modifying, and deleting access rules of file systems ● Creating, querying, and deleting file system tags ● Expanding and shrinking the capacity of a file system ● Querying availability zones ● Read-only permissions on all cloud services if the **Tenant Guest** policy is assigned	System-defined role	Tenant Guest role needs to be assigned in the same project.

Table 1-5 lists all the system-defined roles and policies supported by SFS Turbo.

Table 1-5 System-defined roles and policies supported by SFS Turbo

Role/Policy Name	Description	Type	Dependency
SFS Turbo FullAccess	Administrator permissions for SFS Turbo. Users granted these permissions can perform all operations on SFS Turbo file systems.	System-defined policy	None
SFS Turbo ReadOnlyAccess	Read-only permissions for SFS Turbo. Users granted these permissions can only view SFS Turbo file system data.	System-defined policy	None

Table 1-6 lists the common operations supported by each system-defined policy or role of SFS. Select the policies or roles as required.

Table 1-6 Common operations supported by each system-defined policy or role of SFS

Operation	SFS FullAccess	SFS ReadOnlyAccess	SFS Administrator
Creating a file system	√	x	√
Querying a file system	√	√	√
Modifying a file system	√	x	√
Deleting a file system	√	x	√
Adding an access rule of a file system (Adding a VPC or adding an authorized address to a file system)	√	x	√

Operation	SFS FullAccess	SFS ReadOnlyAccess	SFS Administrator
Modifying an access rule of a file system (Modifying the VPC or authorized address of a file system).	√	x	√
Deleting an access rule of a file system (Deleting the VPC or authorized address of a file system).	√	x	√
Expanding the capacity of a file system	√	x	√
Shrinking the capacity of a file system	√	x	√
Creating file system tags	√	x	√
Querying file system tags	√	√	√
Deleting file system tags	√	x	√
Querying availability zones	√	√	√

1.10 Supported Operating Systems

Table 1-7 lists the operating systems that have passed the compatibility test.

Table 1-7 Supported operating systems

Type	Version
CentOS	CentOS 5, 6, and 7 for x86
Debian	Debian GNU/Linux 6, 7, 8, and 9 for x86
Oracle	Oracle Enterprise Linux 5, 6, and 7 for x86
Red Hat	Red Hat Enterprise Linux 5, 6, and 7 for x86
SUSE	SUSE Linux Enterprise Server 10, 11, and 12 for x86

Type	Version
Ubuntu	Ubuntu 10, 11, 12, 13, 14, and 15 LTS for x86
EulerOS	EulerOS 2
Fedora	Fedora 24 and 25
OpenSUSE	OpenSUSE 42
Windows	Windows Server 2008, 2008 r2, 2012, 2012 r2, and 2016 for x64 Windows 7, 8, and 10

2 Getting Started

2.1 Overview

This section describes how to use SFS.

After creating a file system, you cannot directly access the file system. Instead, you need to mount the file system to ECSs.

Figure 2-1 shows the process for creating and mounting an SFS Turbo file system.

Figure 2-2 shows the process for creating and mounting an SFS Capacity-Oriented file system.

Figure 2-1 Process for using SFS Turbo

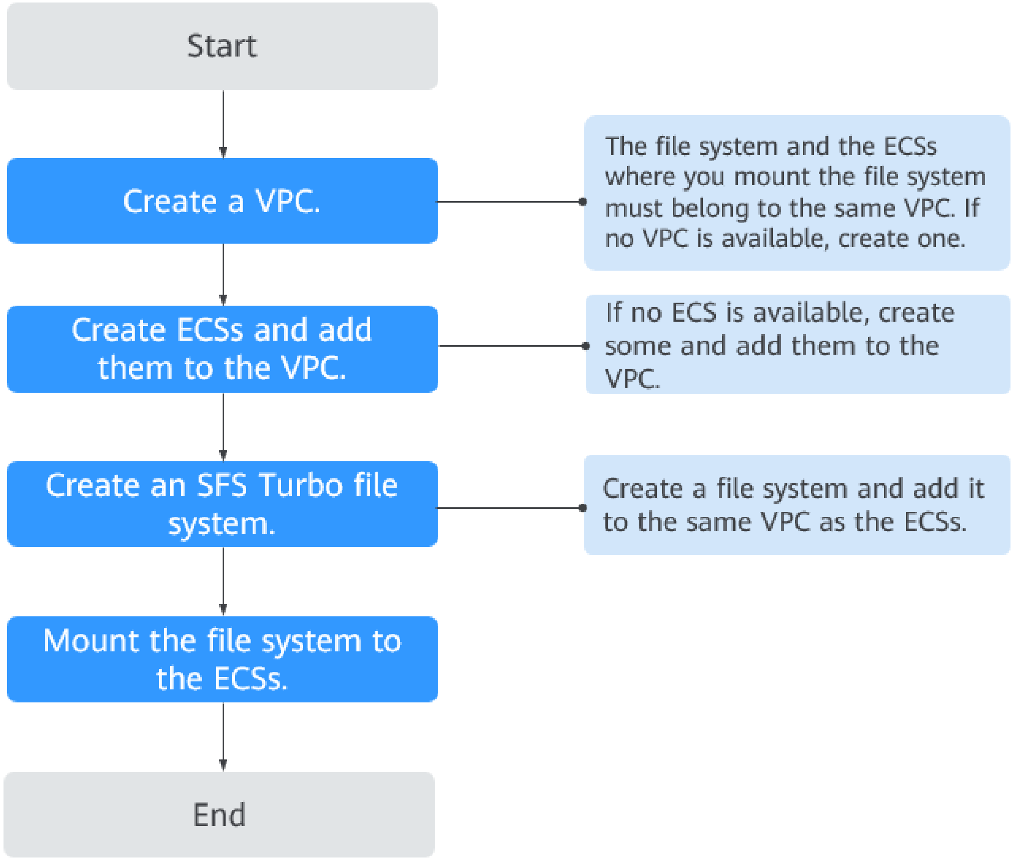
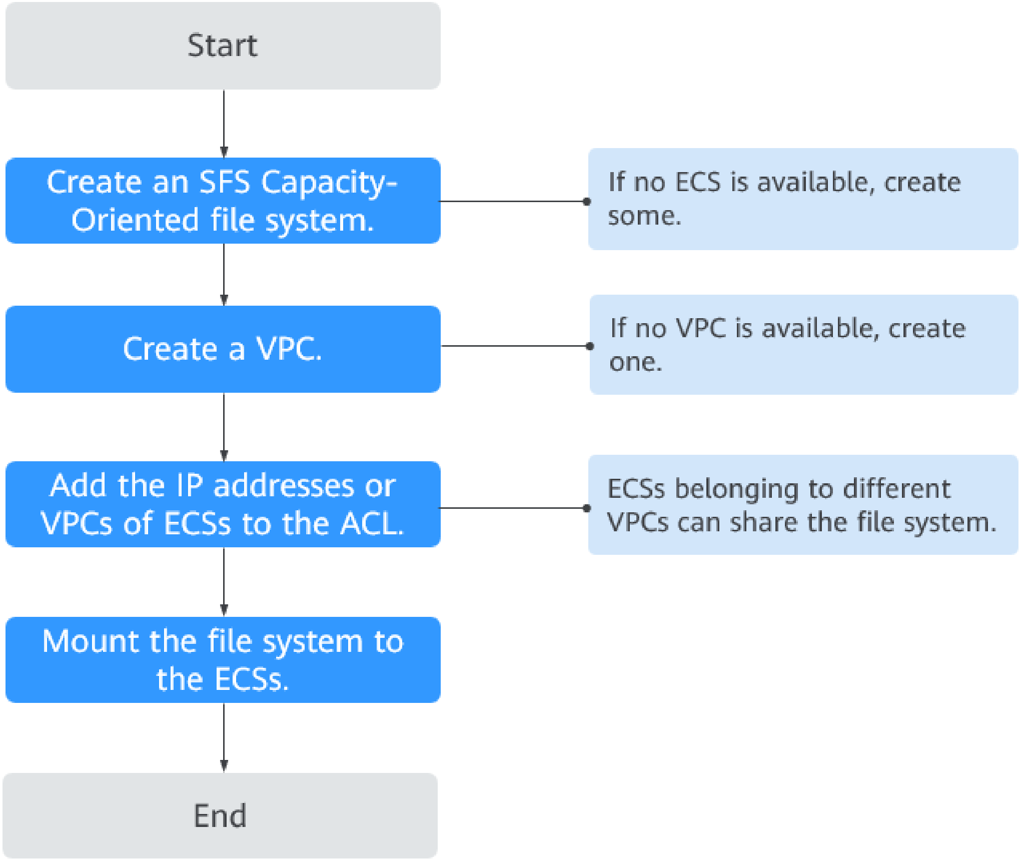

Start
Create a VPC.
The file system and the ECSs where you mount the file system must belong to the same VPC. If no VPC is available, create one.
Create ECSs and add them to the VPC.
If no ECS is available, create some and add them to the VPC.
Create an SFS Turbo file system.
Create a file system and add it to the same VPC as the ECSs.
Mount the file system to the ECSs.
End

Figure 2-2 Process for using SFS Capacity-Oriented


Start
Create an SFS Capacity-Oriented file system.
If no ECS is available, create some.
Create a VPC.
If no VPC is available, create one.
Add the IP addresses or VPCs of ECSs to the ACL.
ECSs belonging to different VPCs can share the file system.
Mount the file system to the ECSs.
End

2.2 Create a File System

You can create a file system and mount it to multiple servers. Then the servers can share this file system. You can create two types of file systems: SFS Capacity-Oriented and SFS Turbo.

Prerequisites

1. Before creating a file system, ensure that a VPC is available.

   If no VPC is available, create one by referring to section "Creating a VPC" in the *Virtual Private Cloud User Guide*.

2. Before creating a file system, ensure that ECSs are available and reside within the created VPC.

   If no ECS is available, create an ECS by referring to "Creating an ECS" in the *Elastic Cloud Server User Guide*.

Creating an SFS Capacity-Oriented File System

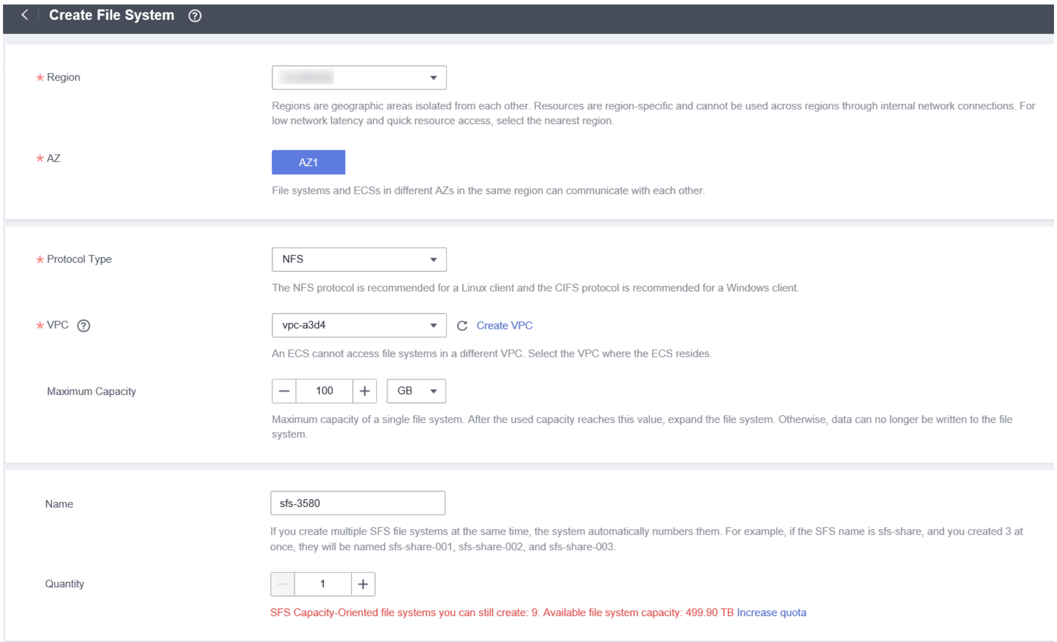
Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Choose **Storage** > **Scalable File Service**.

Step 2 In the navigation pane, choose **SFS Capacity-Oriented**. In the upper right corner of the page, click **Create File System**.

Step 3 Set the parameters as described in **Table 2-1** as shown in **Figure 2-3**.

Figure 2-3 Creating a file system



Create File System

* Region

Regions are geographic areas isolated from each other. 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: AZ1

File systems and ECSs in different AZs in the same region can communicate with each other.

* Protocol Type: NFS

The NFS protocol is recommended for a Linux client and the CIFS protocol is recommended for a Windows client.

* VPC: vpc-a3d4

An ECS cannot access file systems in a different VPC. Select the VPC where the ECS resides.

Maximum Capacity: 100 GB

Maximum capacity of a single file system. After the used capacity reaches this value, expand the file system. Otherwise, data can no longer be written to the file system.

Name: sfs-3580

If you create multiple SFS file systems at the same time, the system automatically numbers them. For example, if the SFS name is sfs-share, and you created 3 at once, they will be named sfs-share-001, sfs-share-002, and sfs-share-003.

Quantity: 1

SFS Capacity-Oriented file systems you can still create: 9. Available file system capacity: 499.90 TB Increase quota

Table 2-1 Parameter description

Parameter	Description	Remarks
AZ	A geographical area with an independent network and an independent power supply.	You are advised to select the same AZ as that of the ECSs.
Protocol Type	SFS supports NFS (only the NFSv3 protocol currently) and CIFS for file system access. The NFS protocol is applicable to Linux ECSs, and the CIFS protocol is applicable to Windows ECSs.	Set this parameter based on site requirements.
VPC	An ECS cannot access file systems in a different VPC. Select the VPC to which the ECS belongs. **NOTE** ● By default, all ECSs in a VPC have the same rights. You can modify the VPC in the future. ● Upon creation, only one VPC can be added for each file system. After a file system is created, you can configure multiple VPCs by referring to **Configuring Multi-VPC Access** for the SFS file system.	Click **View VPC** to view existing VPCs or create a new one.

Parameter	Description	Remarks
Auto Capacity Expansion	Specifies whether the capacity of a file system is limited.	If this function is enabled, the capacity of the file system is not limited. Therefore, you do not need to adjust the capacity of the file system. If this function is disabled, you can set the capacity of the file system and resize the file system later as required. **NOTE** SFS file systems support resizing if the auto capacity expansion function is disabled. You can only enable the auto capacity expansion function when creating a file system. After auto capacity expansion is enabled, you cannot reset the maximum capacity. In addition, auto capacity expansion cannot be disabled after being enabled. Exercise caution when you enable this function.
Maximum Capacity	The maximum capacity of a single file system needs to be configured when the auto capacity expansion function is disabled. When the used capacity of a file system reaches this value, no more data can be written to the file system. You need to expand the file system.	The value ranges from **1 GB** to **512,000 GB**.

Parameter	Description	Remarks
Name	User-defined name of the file system. If you create more than one file system, a name suffix is added to each file system name automatically. For example, if you set the name to **sfs-name** for two new file systems, the two file system names will be **sfs-name-001** and **sfs-name-002**.	The name can contain only letters, digits, underscores (_), and hyphens (-). When creating one file system, enter a maximum of 255 characters. When creating multiple file systems, enter 1 to 251 characters.
Quantity	Number of file systems to be created	Each cloud account can have a total of 512,000 GB for its file systems. Each cloud account can create a maximum of 20 file systems, one by one or in a batch.

Step 4 Click **Create Now**.

Step 5 Confirm the file system information and click **Submit**.

Step 6 Go back to the file system list.

If the status of the created file system is **Available**, the file system is created successfully. If the status is **Creation failed**, contact the administrator.

----End

Creating an SFS Turbo File System

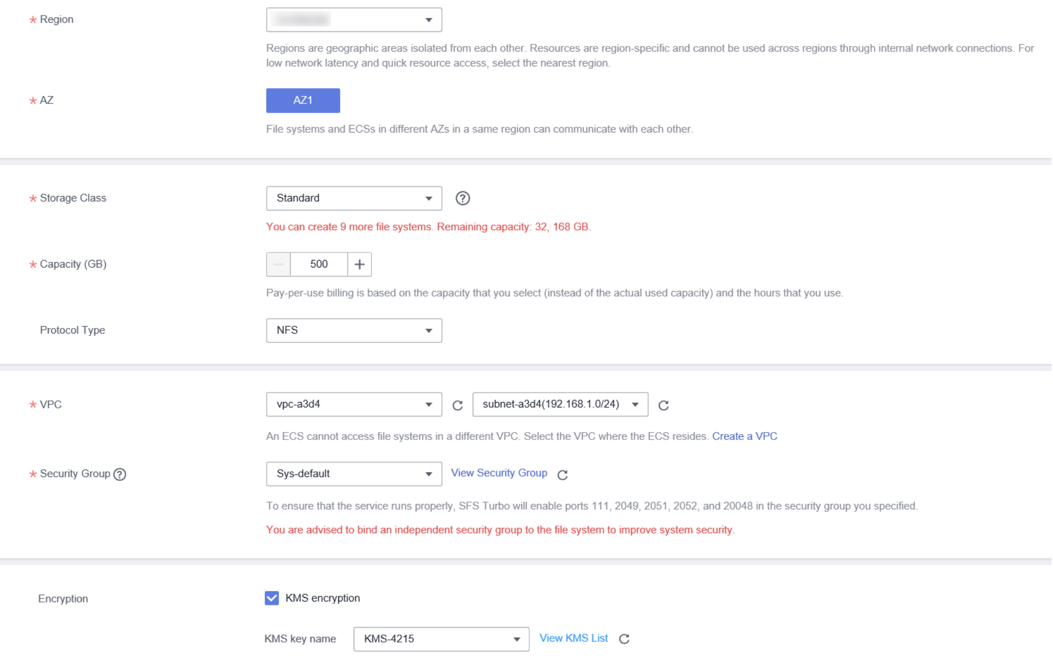
Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Choose **Storage** > **Scalable File Service**.

Step 2 In the navigation pane, choose **SFS Turbo**. In the upper right corner of the page, click **Create File System**.

Step 3 Set the parameters on the page shown in **Figure 2-4**. **Table 2-2** describes the parameters.

Figure 2-4 Creating an SFS Turbo file system



* Region

Regions are geographic areas isolated from each other. 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: AZ1

File systems and ECSs in different AZs in a same region can communicate with each other.

* Storage Class: Standard

You can create 9 more file systems. Remaining capacity: 32, 168 GB.

* Capacity (GB): 500

Pay-per-use billing is based on the capacity that you select (instead of the actual used capacity) and the hours that you use.

Protocol Type: NFS

* VPC: vpc-a3d4 subnet-a3d4(192.168.1.0/24)

An ECS cannot access file systems in a different VPC. Select the VPC where the ECS resides. Create a VPC

* Security Group: Sys-default View Security Group

To ensure that the service runs properly, SFS Turbo will enable ports 111, 2049, 2051, 2052, and 20048 in the security group you specified.

You are advised to bind an independent security group to the file system to improve system security.

Encryption: ☑ KMS encryption

KMS key name: KMS-4215 View KMS List

Table 2-2 Parameter description

Parameter	Description	Remarks
File System Type	Mandatory Select **SFS Capacity-Oriented** or **SFS Turbo**.	Select **SFS Turbo**.
Region	Mandatory Region of the tenant. Select the region from the drop-down list in the upper left corner of the page.	You are advised to select the same region as that of the servers.
AZ	Mandatory A geographical area with an independent network and an independent power supply.	You are advised to select the same AZ as that of the servers.
Protocol Type	Mandatory SFS Turbo supports NFS for file system access.	The default value is **NFS**.

Parameter	Description	Remarks
Storage Class	Mandatory Includes SFS Turbo Standard, SFS Turbo Standard – Enhanced, SFS Turbo Performance, and SFS Turbo Performance – Enhanced. For details about the features and application scenarios of each storage class, see **File System Types**.	Select **Standard**. **NOTE** Once a file system is created, its storage class cannot be changed. If you want to change the storage class, you need to create another file system. Therefore, you are advised to plan the storage class carefully in advance.
Capacity	Maximum capacity of a single file system. When the used capacity of a file system reaches this value, no more data can be written to the file system. You need to expand the file system. The capacity of an SFS Turbo file system cannot be decreased. Set an appropriate file system capacity based on your service needs.	Supported scope: ● SFS Turbo Standard: 500 GB to 32 TB ● SFS Turbo Performance: 500 GB to 32 TB ● SFS Turbo Standard - Enhanced and SFS Turbo Performance - Enhanced: 10 TB to 320 TB.
VPC	Mandatory Select a VPC and its subnet. ● VPC: A server cannot access file systems in a different VPC. Select the VPC to which the server belongs. ● Subnet: A subnet is an IP address range in a VPC. In a VPC, a subnet segment must be unique. A subnet provides dedicated network resources that are logically isolated from other networks, improving network security. **NOTE** Upon creation, only one VPC can be added for each file system. Multi-VPC file sharing can be implemented through VPC peering connection. For details about VPC peering connection, see section "VPC Peering Connection" in *Virtual Private Cloud User Guide*.	-

Parameter	Description	Remarks
Security Group	Mandatory A security group is a virtual firewall that provides secure network access control policies for file systems. You can define different access rules for a security group to protect the file systems that are added to this security group. When creating an SFS Turbo file system, you can select only one security group. You are advised to use an independent security group for an SFS Turbo file system to isolate it from service nodes. The security group rule configuration affects the normal access and use of SFS Turbo. For details about how to configure a security group rule, see section "Adding a Security Group Rule" in the *Virtual Private Cloud User Guide*. After an SFS Turbo file system is created, the system automatically enables the security group port required by the NFS protocol in the SFS Turbo file system. This ensures that the SFS Turbo file system can be accessed by your ECS and prevents file system mounting failures. The inbound ports required by the NFS protocol are ports 111, 445, 2049, 2051, 2052, and 20048. If you need to change the enabled ports, choose **Access Control** > **Security Groups** of the VPC console and locate the target security group.	-

Parameter	Description	Remarks
Encryption	Optional This parameter specifies whether a file system is encrypted. You can create a file system that is encrypted or not, but you cannot change the encryption settings of an existing file system. If **Encryption** is selected, the following parameters will be displayed: ● **KMS key name** **KMS key name** is the identifier of the key, and you can use **KMS key name** to specify the KMS key that is to be used for encryption. Select an existing key from the drop-down list, or click **View KMS List** to create a new key. For details, see "Creating a CMK" in the *Key Management Service User Guide*. ● **KMS key ID** After you select a key name, the system automatically generates a key ID.	-

Parameter	Description	Remarks
Cloud Backup and Recovery	CBR provides backup protection for SFS Turbo and allows you to use backup data to create an SFS Turbo file system. After you set **Cloud Backup and Recovery**, the system binds the SFS Turbo file system to the cloud backup vault and associates the file system with the selected backup policy to periodically back up the file system. The following options are available, among which the default value is **Do not use**: ● **Auto assign**: 1. Set the name of the cloud backup vault, which is a character string consisting of 1 to 64 characters, including letters, digits, underscores (_), and hyphens (-), for example, **vault-f61e**. The default naming rule is **vault_***xxxx*. 2. Enter the vault capacity, which is required for backing up the SFS Turbo file system. The vault capacity cannot be less than the size of the file system. Its value ranges from the total size of the associated file systems to 10,485,760 in the unit of GB. 3. Select a backup policy from the drop-down list, or log in to the CBR console and configure a desired one. ● **Use existing vault**: 1. Select an existing cloud backup vault from the drop-down list. 2. Select a backup policy from the drop-down list, or log in to the CBR console and configure a desired one. ● **Do not use**: Skip this configuration if backup is not	-

Parameter	Description	Remarks
	required. If you need backup protection after a file system has been created, log in to CBR Console, locate the desired vault, and associate the file system to the vault.	
Enterprise Project	This parameter is provided for enterprise users. When creating a file system, you can add the file system to an existing enterprise project. An enterprise project is a cloud resource management mode, in which cloud resources and members are centrally managed by project. The default project is **default**. Select an enterprise project from the drop-down list.	You can select only created enterprise projects. To create an enterprise project, click **Enterprise** in the upper right corner of the console page.
Name	Mandatory User-defined name of the file system.	The value can contain only letters, digits, and hyphens (-). The name of the created file system must contain more than four characters and less than or equal to 64 characters.

Step 4 Click **Create Now**.

Step 5 Confirm the file system information and click **Submit**.

Step 6 Complete the creation and go back to the file system list.

If the status of the created file system is **Available**, the file system is created successfully. If the status is **Creation failed**, contact the administrator.

----End

2.3 Mount a File System

2.3.1 Mounting an NFS File System to ECSs (Linux)

After creating a file system, you need to mount the file system to servers so that they can share the file system.

CIFS file systems cannot be mounted to Linux .

An SFS Capacity-Oriented file system can use either NFS or CIFS. It cannot use both protocols.

In this section, ECSs are used as example servers. Operations on BMSs and containers (CCE) are the same as those on ECSs.

To use SFS Turbo as the storage backend for CCE, see section "Storage" or "Storage (FlexVolume)" in the *Cloud Container Engine User Guide*. Then complete the deployment on the CCE console.

Prerequisites

- You have checked the type of the operating system on each ECS. Different operating systems use different commands to install the NFS client.
- You have created a file system and have obtained the mount point of the file system.
- At least one ECS that belongs to the same VPC as the file system exists.
- The IP address of the DNS server for resolving the domain names of the file systems has been configured on the ECS. SFS Turbo file systems do not require domain name resolution.

Procedure

Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to go to the ECS console.

Step 2 Log in to the ECS as user **root**.

NOTE

If you log in to the ECS as a non-root user, see **Mounting a File System to a Linux ECS as a Non-root User**.

Step 3 Install the NFS client.

1. Run the following command to check whether the NFS software package is installed.
   - On CentOS, Red Hat, Oracle Enterprise Linux, SUSE, EulerOS, Fedora, or OpenSUSE:

     rpm -qa|grep nfs
   - On Debian or Ubuntu:

     dpkg -l nfs-common

   If a command output similar to the following is displayed, the NFS software package has been installed and you can go to **Step 4**. If nothing is displayed, go to **Step 3.2**.
   - On CentOS, Red Hat, EulerOS, Fedora, or Oracle Enterprise Linux:
     ```
     libnfsidmap
     nfs-utils
     ```
   - On SUSE or OpenSUSE:
     ```
     nfsidmap
     nfs-client
     ```
   - On Debian or Ubuntu:
     ```
     nfs-common
     ```
2. Run the following command to install the NFS software package.

NOTE

The following commands require that ECSs be connected to the Internet. Or, the installation will fail.

- On CentOS, Red Hat, EulerOS, Fedora, or Oracle Enterprise Linux:

  sudo yum -y install nfs-utils
- On Debian or Ubuntu:

  sudo apt-get install nfs-common
- On SUSE or OpenSUSE:

  zypper install nfs-client

Step 4 Run the following command to check whether the domain name in the file system mount point can be resolved.

nslookup *File system domain name*

NOTE

- A file system domain name is just a part of the mount point, for example, **sfs-nas1.***xxxx***.com**. You can obtain a file system domain name from the mount point of a file system. In this step, you are not supposed to enter the entire mount point but only the domain name.
- If the **nslookup** command cannot be used, install the **bind-utils** software package by running the **yum install bind-utils** command.

- If the domain name can be resolved, go to **Step 5**.
- If the domain name cannot be resolved, configure the DNS server IP address and then mount the file system. For details, see **Configuring DNS**.

Step 5 Run the following command to create a local path for mounting the file system:

mkdir *Local path*

NOTE

If there is any resource, such as a disk, already mounted on the local path, create a new path. (NFS clients do not refuse repeated mounts. If there are repeated mounts, information of the last successful mount is displayed.)

Step 6 Run the following command to mount the file system to the ECS that belongs to the same VPC as the file system. Currently, the file system can be mounted to Linux ECSs using NFSv3 only.

Table 2-3 describes the variables.

To mount an SFS Capacity-Oriented file system, run the following command: **mount -t nfs -o vers=3,timeo=600,noresvport,nolock** *Mount point Local path*

To mount an SFS Turbo file system, run the following command: **mount -t nfs -o vers=3,timeo=600,noresvport,nolock,tcp** *Mount point Local path*

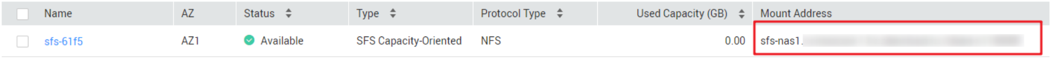
NOTICE

After an ECS where file systems have been mounted restarts, it loses the file system mount information. You can configure automatic mount in the **fstab** file to ensure that an ECS automatically mounts file systems when it restarts. For details, see **Mounting a File System Automatically**.

Table 2-3 Parameter description

Parameter	Description
vers	File system version. Only NFSv3 is supported currently, so the value is fixed to **3**.
timeo	Waiting time before the NFS client retransmits a request. The unit is 0.1 second. The recommended value is **600**.
resvport/ noresvport	Whether the confidential source port is used for server connection. By default, **resvport** indicates that the confidential port is used, and **noresvport** indicates that the confidential port is not used. This parameter is supported by Linux kernel 2.6.28 or later. You are advised to set this parameter to **noresvport**, which can tell NFS clients to use a TCP source port when reconnecting to the network, thereby ensuring the continuous availability of the SFS file system in the event of a network failure.
lock/nolock	Whether to lock files on the server using the NLM protocol. If **nolock** is selected, the lock is valid for applications on one host. For applications on another host, the lock is invalid. The recommended value is **nolock**. If this parameter is not specified, **lock** is selected by default. In this case, other servers cannot write data to the file system.
Mount point	The format for an SFS Capacity-Oriented file system is *File system domain name*:/*Path*, for example, **example.com:/ share-***xxx*. The format for an SFS Turbo file system is *File system IP address*:/, for example, **192.168.0.0:/**. See **Figure 2-5**. **NOTE** ● *x* is a digit or letter. ● If the mount point is too long to display completely, you can adjust the column width. ● Hover the mouse over the mount point to display the complete **mount** command.
Local path	Local path on the ECS, used to mount the file system, for example, **/local_path**.

Figure 2-5 Mount point



□	Name	AZ	Status	Type	Protocol Type	Used Capacity (GB)	Mount Address
□	sfs-61f5	AZ1	● Available	SFS Capacity-Oriented	NFS	0.00	sfs-nas1.[redacted]

For more mounting parameters for performance optimization during file system mounting, see **Table 2-4**. Use commas (,) to separate parameters. The following command is an example:

mount -t nfs -o vers=3,timeo=600,nolock,rsize=1048576,wsize=1048576,hard,retrans=3,noresvport,ro,async,noatime,nodiratime *Mount point Local path*

Table 2-4 Parameters for file system mounting

Parameter	Description
rsize	Maximum number of bytes that can be read from the server each time. The actual data is less than or equal to the value of this parameter. The value of **rsize** must be a positive integer that is a multiple of **1024**. If the entered value is smaller than **1024**, the value is automatically set to **4096**. If the entered value is greater than **1048576**, the value is automatically set to **1048576**. By default, the setting is performed after the negotiation between the server and the client. You are advised to set this parameter to the maximum value **1048576**.
wsize	Maximum number of bytes that can be written to the server each time. The actual data is less than or equal to the value of this parameter. The value of **wsize** must be a positive integer that is a multiple of **1024**. If the entered value is smaller than **1024**, the value is automatically set to **4096**. If the entered value is greater than **1048576**, the value is automatically set to **1048576**. By default, the setting is performed after the negotiation between the server and the client. You are advised to set this parameter to the maximum value **1048576**.
soft/hard	**soft** indicates that a file system is mounted in soft mount mode. In this mode, if an NFS request times out, the client returns an error to the invoking program. **hard** indicates that a file system is mounted in hard mount mode. In this mode, if the NFS request times out, the client continues to request until the request is successful. The default value is **hard**.
retrans	Number of retransmission times before the client returns an error.

Parameter	Description
ro/rw	● **ro**: indicates that the file system is mounted as read-only. ● **rw**: indicates that the file system is mounted as read/write. The default value is **rw**. If this parameter is not specified, the file system will be mounted as read/write.
resvport/noresvport	Whether the confidential source port is used for server connection. By default, **resvport** indicates that the confidential port is used, and **noresvport** indicates that the confidential port is not used. This parameter is supported by Linux kernel 2.6.28 or later. You are advised to set this parameter to **noresvport**, which can tell NFS clients to use a TCP source port when reconnecting to the network, thereby ensuring the continuous availability of the SFS file system in the event of a network failure.
sync/async	**sync** indicates that data is written to the server immediately. **async** indicates that data is first written to the cache before being written to the server. Synchronous write requires that an NFS server returns a success message only after all data is written to the server, which brings long latency. The recommended value is **async**.
noatime	If you do not need to record the file access time, set this parameter. This prevents overheads caused by access time modification during frequent access.
nodiratime	If you do not need to record the directory access time, set this parameter. This prevents overheads caused by access time modification during frequent access.

NOTE

You are advised to use the default values for the parameters without usage recommendations.

Step 7 Run the following command to view the mounted file system:

mount -l

If the command output contains the following information, the file system has been mounted.

```
Mount point on /local_path type nfs (rw,vers=3,timeo=600,nolock,addr=)
```

Step 8 After the file system is mounted successfully, access the file system on the ECSs to read or write data.

If the mounting fails or times out, rectify the fault by referring to **Troubleshooting**.

NOTE

The maximum size of a file that can be written to an SFS Capacity-Oriented file system is 240 TB.

The maximum size of a file that can be written to an SFS Turbo file system is 32 TB, and that for an SFS Turbo Enhanced file system is 320 TB.

----End

2.3.2 Mounting an NFS File System to ECSs (Windows)

After creating a file system, you need to mount the file system to servers so that they can share the file system.

This section uses Windows Server 2012 as the example OS to describe how to mount an NFS file system. For other versions, perform the steps based on the actual situation.

An SFS Capacity-Oriented file system can support either the NFS or CIFS protocol.

In this section, ECSs are used as example servers. Operations on BMSs and containers (CCE) are the same as those on ECSs.

Prerequisites

- You have created a file system and have obtained the mount point of the file system.
- At least one ECS that belongs to the same VPC as the file system exists.
- The IP address of the DNS server for resolving the domain names of the file systems has been configured on the ECS. For details, see **Configuring DNS**. SFS Turbo file systems do not require domain name resolution.

Limitations and Constraints

You are advised to use CIFS file systems in Windows OS.

Procedure

Step 1 Log in to the management console using a cloud account.

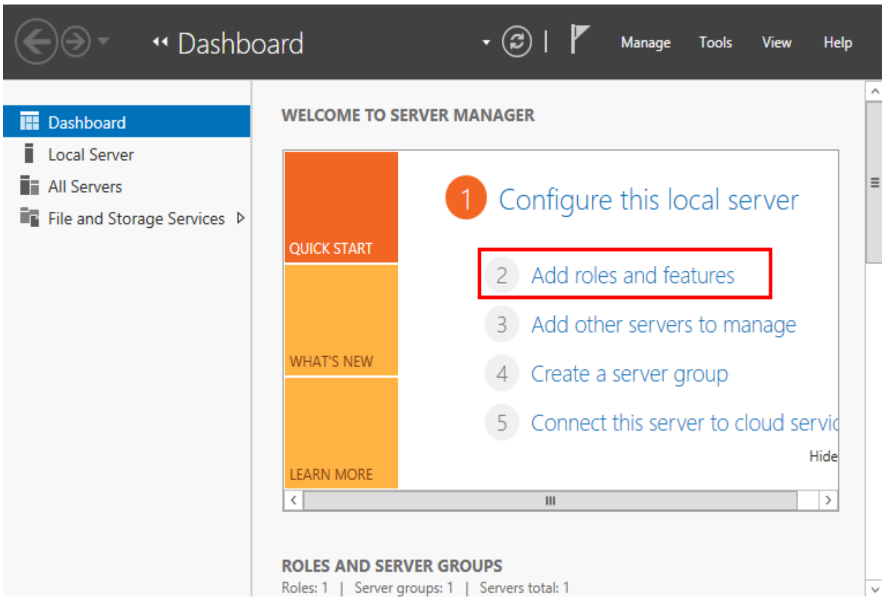
1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to switch to the ECS console.

Step 2 Go to the ECS console and log in to the ECS running Windows Server 2012.

Step 3 Install the NFS client.

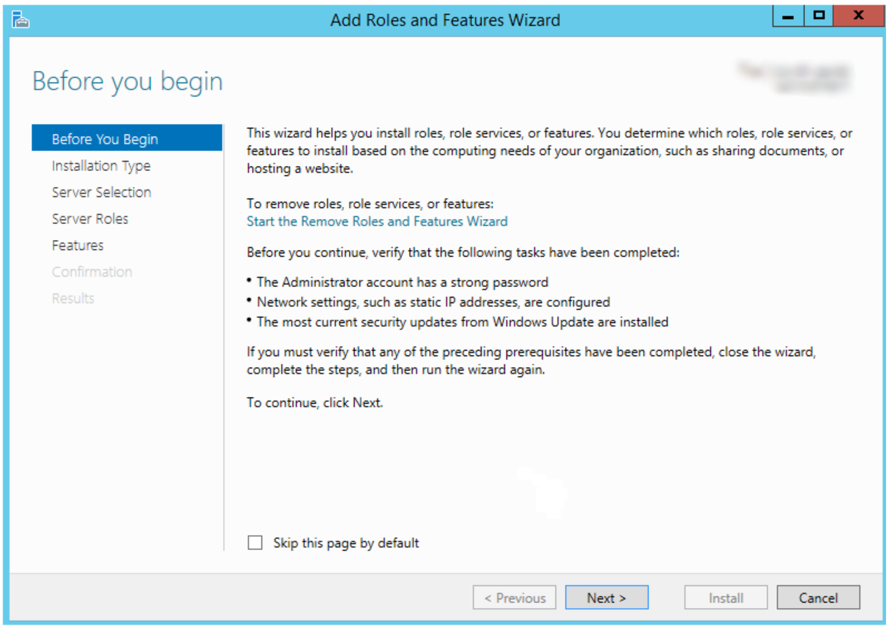
1. Click **Server Manager** in the lower left corner. The **Server Manager** window is displayed, as shown in **Figure 2-6**.

Figure 2-6 Server Manager


Dashboard
Manage
Tools
View
Help
Dashboard
Local Server
All Servers
File and Storage Services
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
LEARN MORE
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group
5 Connect this server to cloud servic
Hide
ROLES AND SERVER GROUPS
Roles: 1 | Server groups: 1 | Servers total: 1

2. Click **Add Roles and Features**. See **Figure 2-7**.

Figure 2-7 Wizard for adding roles and features



This wizard helps you install roles, role services, or features. You determine which roles, role services, or features to install based on the computing needs of your organization, such as sharing documents, or hosting a website.

To remove roles, role services, or features:
Start the Remove Roles and Features Wizard

Before you continue, verify that the following tasks have been completed:

- The Administrator account has a strong password
- Network settings, such as static IP addresses, are configured
- The most current security updates from Windows Update are installed

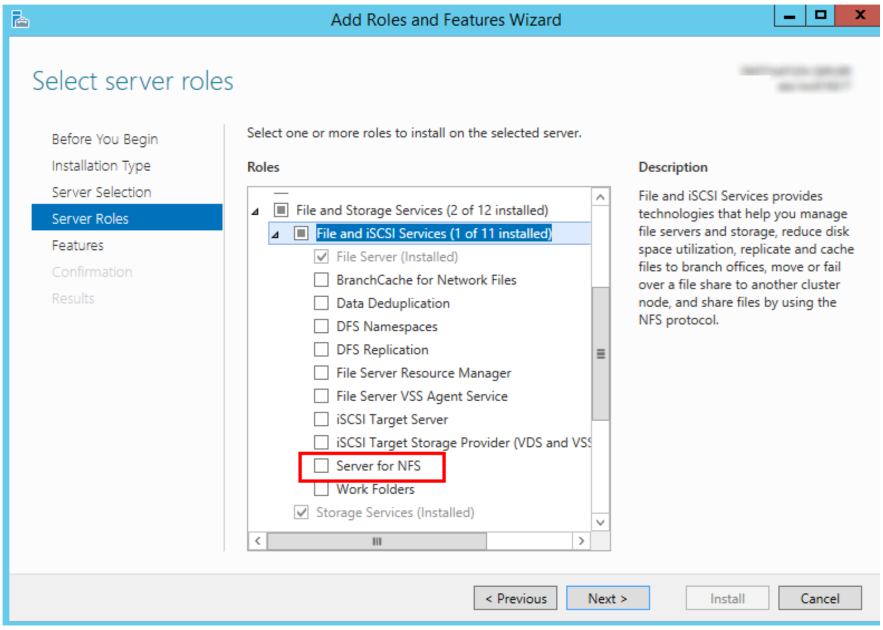
If you must verify that any of the preceding prerequisites have been completed, close the wizard, complete the steps, and then run the wizard again.

To continue, click Next.

□ Skip this page by default

3. Click **Next** as prompted. On the **Server Roles** page, select **Server for NFS**, as shown in **Figure 2-8**.

Figure 2-8 Selecting the server for NFS



Add Roles and Features Wizard

Select server roles

[redacted]

Select one or more roles to install on the selected server.

Roles

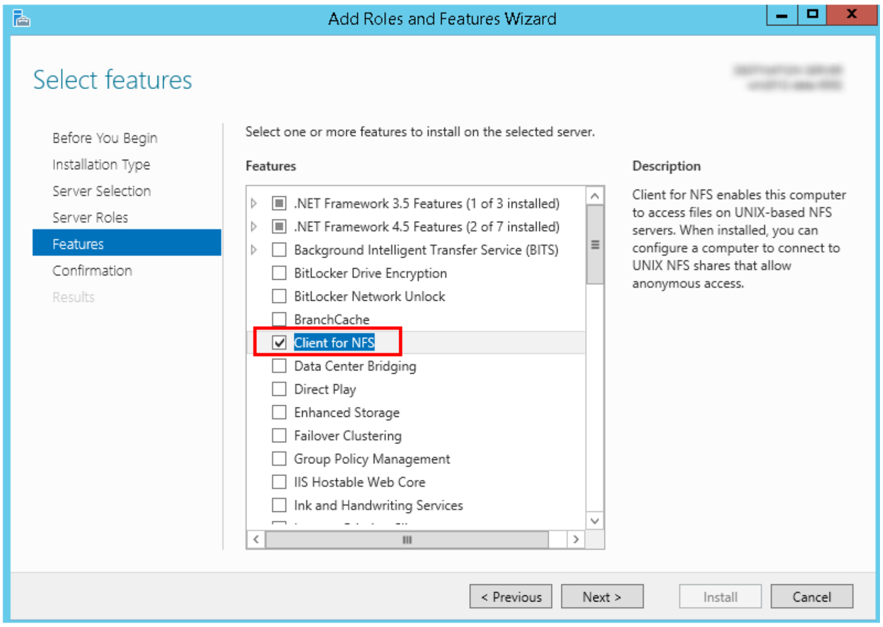
- ◪ File and Storage Services (2 of 12 installed)
  - ◪ File and iSCSI Services (1 of 11 installed)
    - ☑ File Server (Installed)
    - □ BranchCache for Network Files
    - □ Data Deduplication
    - □ DFS Namespaces
    - □ DFS Replication
    - □ File Server Resource Manager
    - □ File Server VSS Agent Service
    - □ iSCSI Target Server
    - □ iSCSI Target Storage Provider (VDS and VSS
    - □ Server for NFS
    - □ Work Folders
  - ☑ Storage Services (Installed)

Description

File and iSCSI Services provides technologies that help you manage file servers and storage, reduce disk space utilization, replicate and cache files to branch offices, move or fail over a file share to another cluster node, and share files by using the NFS protocol.

4. Click **Next**. In the **Features** page, select **Client for NFS** and click **Next**, as shown in Figure 2-9. Confirm the settings and then click **Install**. If you install the NFS client for the first time, after the installation is complete, restart the client and log in to the ECS again as prompted.

Figure 2-9 Selecting the NFS client



Add Roles and Features Wizard

Select features

[redacted]

Select one or more features to install on the selected server.

Features

- ◪ .NET Framework 3.5 Features (1 of 3 installed)
- ◪ .NET Framework 4.5 Features (2 of 7 installed)
- □ Background Intelligent Transfer Service (BITS)
- □ BitLocker Drive Encryption
- □ BitLocker Network Unlock
- □ BranchCache
- ☑ Client for NFS
- □ Data Center Bridging
- □ Direct Play
- □ Enhanced Storage
- □ Failover Clustering
- □ Group Policy Management
- □ IIS Hostable Web Core
- □ Ink and Handwriting Services

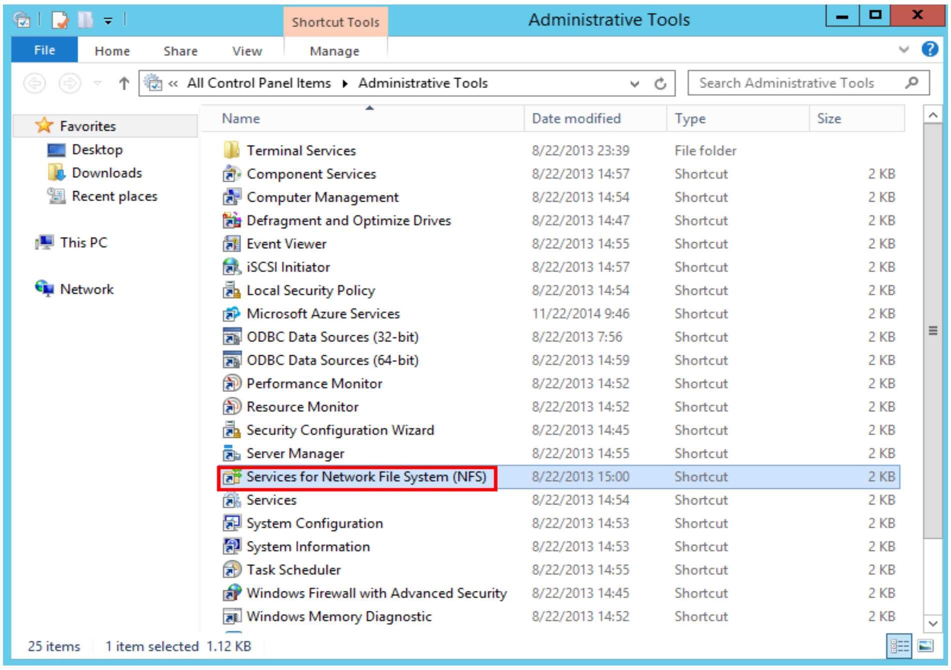
Description

Client for NFS enables this computer to access files on UNIX-based NFS servers. When installed, you can configure a computer to connect to UNIX NFS shares that allow anonymous access.

Step 4 Modify the NFS transfer protocol.

1. Choose **Control Panel > System and Security > Administrative Tools > Services for Network File System (NFS)**, as shown in Figure 2-10.

Figure 2-10 Administrative tools



Name	Date modified	Type	Size
Terminal Services	8/22/2013 23:39	File folder	
Component Services	8/22/2013 14:57	Shortcut	2 KB
Computer Management	8/22/2013 14:54	Shortcut	2 KB
Defragment and Optimize Drives	8/22/2013 14:47	Shortcut	2 KB
Event Viewer	8/22/2013 14:55	Shortcut	2 KB
iSCSI Initiator	8/22/2013 14:57	Shortcut	2 KB
Local Security Policy	8/22/2013 14:54	Shortcut	2 KB
Microsoft Azure Services	11/22/2014 9:46	Shortcut	2 KB
ODBC Data Sources (32-bit)	8/22/2013 7:56	Shortcut	2 KB
ODBC Data Sources (64-bit)	8/22/2013 14:59	Shortcut	2 KB
Performance Monitor	8/22/2013 14:52	Shortcut	2 KB
Resource Monitor	8/22/2013 14:52	Shortcut	2 KB
Security Configuration Wizard	8/22/2013 14:45	Shortcut	2 KB
Server Manager	8/22/2013 14:55	Shortcut	2 KB
Services for Network File System (NFS)	8/22/2013 15:00	Shortcut	2 KB
Services	8/22/2013 14:54	Shortcut	2 KB
System Configuration	8/22/2013 14:53	Shortcut	2 KB
System Information	8/22/2013 14:53	Shortcut	2 KB
Task Scheduler	8/22/2013 14:55	Shortcut	2 KB
Windows Firewall with Advanced Security	8/22/2013 14:45	Shortcut	2 KB
Windows Memory Diagnostic	8/22/2013 14:52	Shortcut	2 KB

25 items 1 item selected 1.12 KB

2. Right-click **Client for NFS**, choose **Properties**, change the transport protocol to **TCP**, and select **Use hard mounts**, as shown in **Figure 2-11** and **Figure 2-12**.

Figure 2-11 Services for NFS

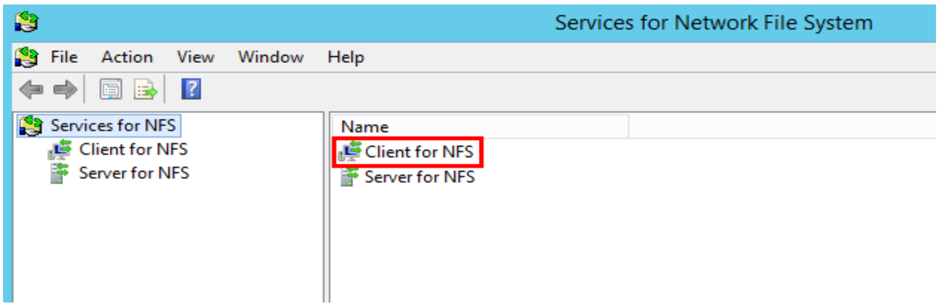
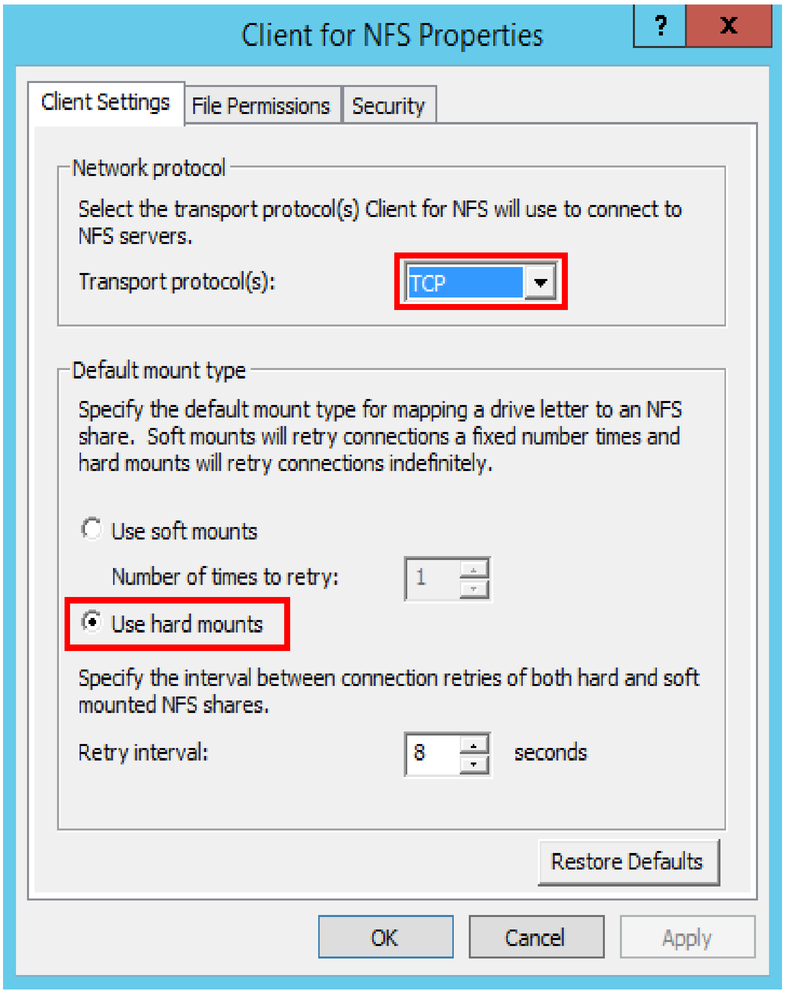


Figure 2-12 Client for NFS properties



Network protocol

Select the transport protocol(s) Client for NFS will use to connect to NFS servers.

Transport protocol(s): TCP

Default mount type

Specify the default mount type for mapping a drive letter to an NFS share. Soft mounts will retry connections a fixed number times and hard mounts will retry connections indefinitely.

○ Use soft mounts

Number of times to retry: 1

● Use hard mounts

Specify the interval between connection retries of both hard and soft mounted NFS shares.

Retry interval: 8 seconds

Step 5 Check that the IP address of the DNS server for resolving the domain names of the file systems has been configured on the ECS before mounting the file system. For details, see **Configuring DNS**. SFS Turbo file systems do not require domain name resolution.

Step 6 Run the following command in the Command Prompt of the Windows Server 2012 (**X** is the drive letter of the free disk). Select the ECS that belongs to the same VPC as the file system to mount the file system.

For SFS Capacity-Oriented file systems: **mount -o nolock** *mount point* **X:**

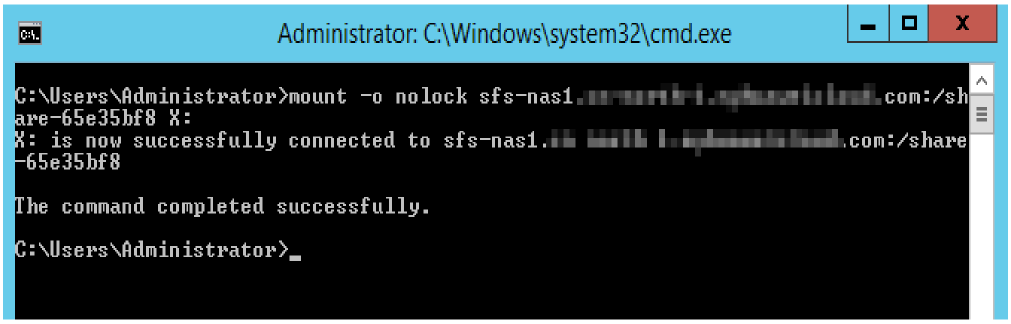
For SFS Turbo file systems: **mount -o nolock -o casesensitive=yes** *IP address*:**/! X:**

NOTE

- Free drive letter of the disk: A drive letter that is not in use, such as driver letter E or X.
- The mount point of an SFS Turbo file system is the root directory. **Ensure that an English exclamation mark (!) is added to the mount point**, for example, **127.0.0.1:/!**.
- **casesensitive=yes** indicates that file names are case sensitive during file search. If this parameter is not added, the performance of creating files in a large directory will deteriorate.

You can move the cursor to the mount point and click next to the mount point to copy the mount point. If the information shown in **Figure 2-13** is displayed, the mounting is successful.

Figure 2-13 Running the command

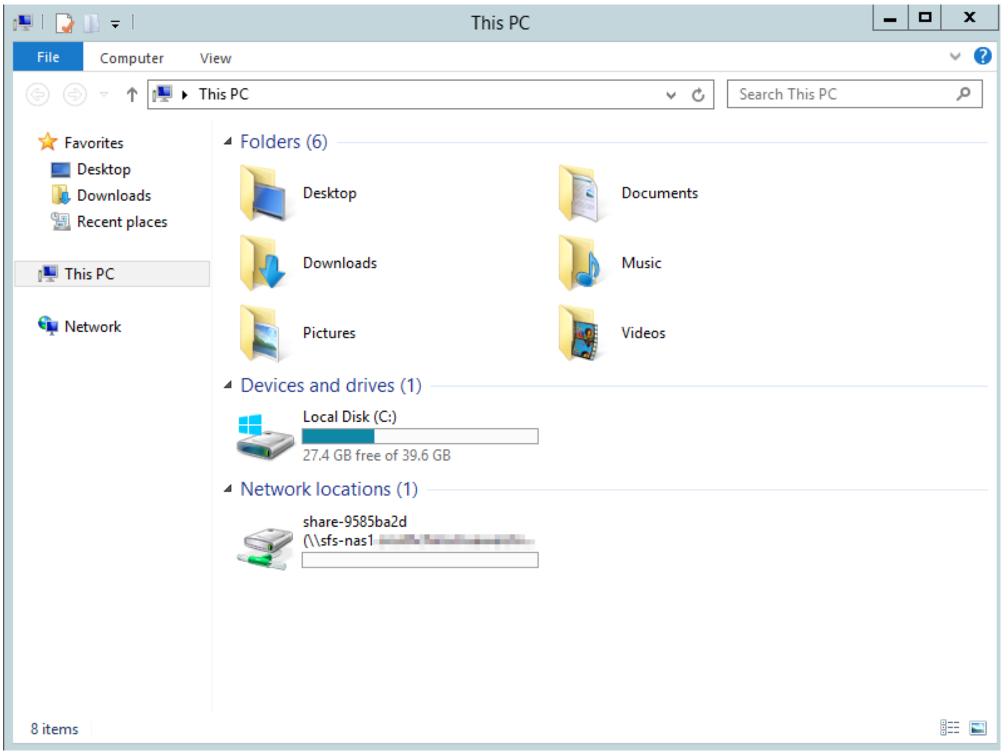


```
C:\Users\Administrator>mount -o nolock sfs-nas1.[redacted].com:/sh
are-65e35bf8 X:
X: is now successfully connected to sfs-nas1.[redacted].com:/share
-65e35bf8

The command completed successfully.

C:\Users\Administrator>_
```

Step 7 After the file system is mounted successfully, you can view the mounted file system on the **This PC** window, as shown in **Figure 2-14**.

If the mounting fails or times out, rectify the fault by referring to **Troubleshooting**.

Figure 2-14 Successful mounting



NOTE

To distinguish different file systems mounted on an ECS, you can rename file systems by right-clicking a file system and choose **Rename**.

----End

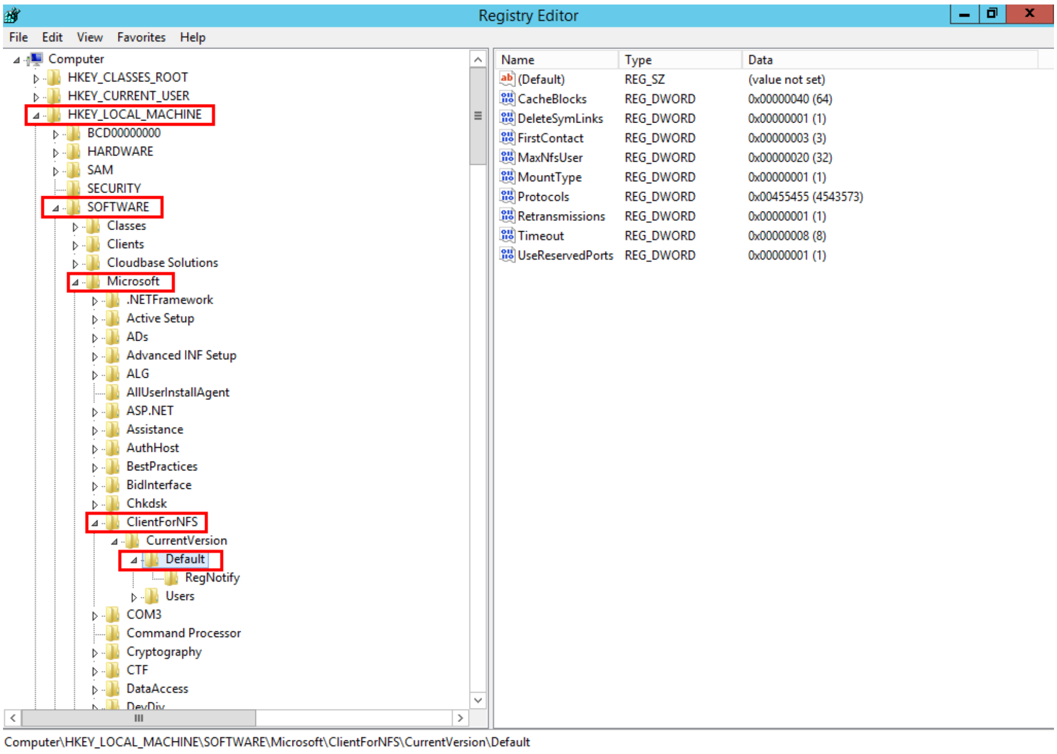
Troubleshooting

If a file system is mounted to a Linux ECS and a Windows ECS, on the Windows ECS, data cannot be written to the files created by the Linux ECS. To address this problem, modify the registry and change both UID and GID values to **0** for NFS accesses from Windows. This section uses Windows Server 2012 as an example. Do as follows:

Step 1 Choose **Start** > **Run** and enter **regedit** to open the registry.

Step 2 Enter the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default** directory. See **Figure 2-15**.

Figure 2-15 Entering the directory

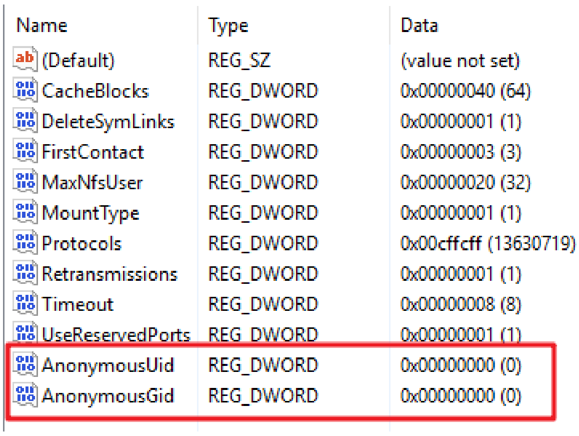


Name	Type	Data
(Default)	REG_SZ	(value not set)
CacheBlocks	REG_DWORD	0x00000040 (64)
DeleteSymLinks	REG_DWORD	0x00000001 (1)
FirstContact	REG_DWORD	0x00000003 (3)
MaxNfsUser	REG_DWORD	0x00000020 (32)
MountType	REG_DWORD	0x00000001 (1)
Protocols	REG_DWORD	0x00455455 (4543573)
Retransmissions	REG_DWORD	0x00000001 (1)
Timeout	REG_DWORD	0x00000008 (8)
UseReservedPorts	REG_DWORD	0x00000001 (1)

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default

Step 3 Right-click the blank area and choose **New** > **DWORD Value** from the shortcut menu. Set **AnonymousUid** and **AnonymousGid** to **0**. **Figure 2-16** shows a successful operation.

Figure 2-16 Adding values



Name	Type	Data
(Default)	REG_SZ	(value not set)
CacheBlocks	REG_DWORD	0x00000040 (64)
DeleteSymLinks	REG_DWORD	0x00000001 (1)
FirstContact	REG_DWORD	0x00000003 (3)
MaxNfsUser	REG_DWORD	0x00000020 (32)
MountType	REG_DWORD	0x00000001 (1)
Protocols	REG_DWORD	0x00cffcff (13630719)
Retransmissions	REG_DWORD	0x00000001 (1)
Timeout	REG_DWORD	0x00000008 (8)
UseReservedPorts	REG_DWORD	0x00000001 (1)
AnonymousUid	REG_DWORD	0x00000000 (0)
AnonymousGid	REG_DWORD	0x00000000 (0)

Step 4 After modifying the registry, restart the server for the modification to take effect.

----End

2.3.3 Mounting a CIFS File System to ECSs (Windows)

After creating a file system, you need to mount the file system to ECSs so that they can share the file system.

This section uses Windows Server 2012 as an example to describe how to mount a CIFS file system.

An SFS Capacity-Oriented file system can support either the NFS or CIFS protocol.

Prerequisites

- You have created a file system and have obtained the mount point of the file system.
- At least one ECS that belongs to the same VPC as the file system exists.
- The IP address of the DNS server for resolving the domain names of the file systems has been configured on the ECSs. For details, see **Configuring DNS**.
- You need to mount the file system as user **Administrator**. You cannot switch to another user to mount the file system.

Limitations and Constraints

CIFS file systems cannot be mounted to Linux ECSs.

Procedure

Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to switch to the ECS console.

Step 2 Go to the ECS console and log in to the ECS running Windows Server 2012.

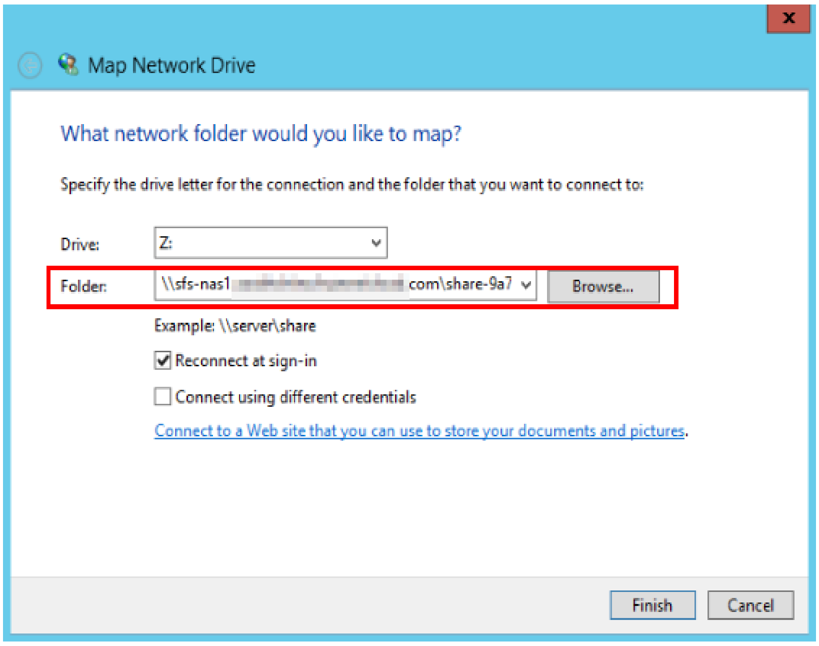
Step 3 Click **Start**, right-click **Computer**, and choose **Map network drive**.

Step 4 In the dialog box that is displayed, enter the mount point of the file system, specifically, \\*File system domain name**Path*. See **Figure 2-17**.

Table 2-5 Variable description

Variable	Description
File system domain name	Obtain the file system domain name from the file system mount point. For details about how to obtain the file system domain name, see **File System Management**.
Path	The format is **share-***xxxxxxxx*, where *x* is a digit or letter.

Figure 2-17 Entering the mount point



Map Network Drive

What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: Z:

Folder: \\sfs-nas1[redacted]com\share-9a7

Example: \\server\share

☑ Reconnect at sign-in

□ Connect using different credentials

Connect to a Web site that you can use to store your documents and pictures.

Step 5 Click **Finish**.

Step 6 After the file system is mounted successfully, you can view the mounted file system on the **This PC** page.

If the mounting fails or times out, rectify the fault by referring to **Troubleshooting**.

----End

2.3.4 Mounting a File System Automatically

File system mounting information may be lost after a server is restarted. You can configure automatic mounting for the server to avoid the mounting information loss.

Restrictions

Because the service startup sequences in different operating systems vary, some servers running CentOS may not support the following automatic mounting schemes. In this case, manually mount the file system.

Procedure (Linux)

Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to switch to the ECS console.

Step 2 Log in to the ECS as user **root**.

Step 3 Run the **vi /etc/fstab** command to edit the **/etc/fstab** file.

At the end of the file, add the file system information, for example:

```
Mount point /local_path nfs vers=3,timeo=600,nolock 0 0
```

Replace *Mount point* and */local_path* with actual values. You can obtain the mount point from the **Mount Address** column of the file system. Each record in the **/etc/fstab** file corresponds to a mount. Each record has six fields, as described in **Field Description**.

NOTICE

For optimal system performance, configure file system information based on the previous example configuration. If needed, you can customize part of mount parameters. However, the customization may affect system performance.

Step 4 Press **Esc**, input **:wq**, and press **Enter** to save and exit.

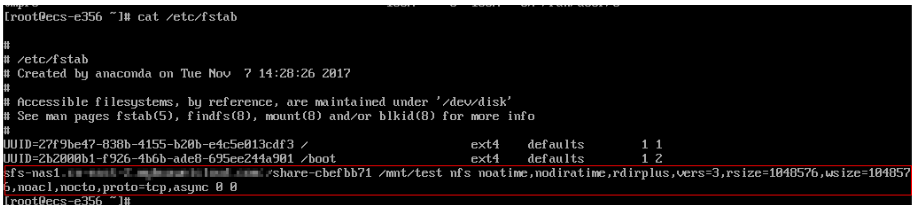
After the preceding configurations are complete, the system reads mounting information from the **/etc/fstab** file to automatically mount the file system when the ECS restarts.

Step 5 (Optional) Run the following command to view the updated content of the **/etc/fstab** file:

cat /etc/fstab

Figure 2-18 shows the updated file content.

Figure 2-18 Updated file content



```
[root@ecs-e356 ~]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Tue Nov  7 14:28:26 2017
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=27f9be47-838b-4155-b20b-e4c5e013cdf3 /                       ext4    defaults        1 1
UUID=2b2000b1-f926-4b6b-ade8-695ee244a901 /boot                   ext4    defaults        1 2
sfs-nas1.[redacted]/share-cbefbb71 /mnt/test nfs noatime,nodiratime,rdirplus,vers=3,rsize=1048576,wsize=104857
6,noacl,nocto,proto=tcp,async 0 0
[root@ecs-e356 ~]#
```

Step 6 If the automatic mounting fails due to a network issue, add the **sleep** parameter and a time in front of the mounting command in the **rc.local** file, and mount the file system after the NFS service is started.

```
sleep 10s && sudo mount -t nfs -o vers=3,timeo=600,noresvport,nolock Mount point/local_path
```

----End

Field Description

Table 1 describes the mount fields.

Table 2-6 Field description

Field	Description
Mount point	Mount object, that is, the mount point of the file system to be mounted. Set this parameter to the mount point in the **mount** command that is used in **Mounting an NFS File System to ECSs (Linux)**.
/local_path	Mount point, that is, the directory created on the ECS for mounting the file system. Set this parameter to the local path in the **mount** command that is used in **Mounting an NFS File System to ECSs (Linux)**.
nfs	Mount type, that is, file system or partition type. Set it to **nfs**.
vers=3,timeo=600,nolock	Mount options, used to set mount parameters. Use commas (,) to separate between multiple options. ● **vers**: file system version. The value **3** indicates NFSv3. ● **timeo**: waiting time before the NFS client retransmits a request. The unit is 0.1 second. The recommended value is **600**. ● **nolock**: specifies whether to lock files on the server using the NLM protocol.
0	Choose whether to back up file systems using the dump command. ● **0**: not to back up file systems ● An integer larger than 0: to back up file systems. A file system with a smaller integer is checked earlier than that with a larger integer.
0	Choose whether to check file systems using the fsck command when the ECS is starting and specify the sequence for checking file systems. ● **0**: to check file systems ● By default, this field is set to **1** for the root directory partition. Other partitions start from **2**, and a partition with a smaller integer is checked earlier than that with a larger integer.

Procedure (Windows)

Ensure that an NFS client has been installed on the target server before mounting. This section uses Windows Server 2012 as an example to describe how to mount a file system.

Step 1 Log in to the management console using a cloud account.

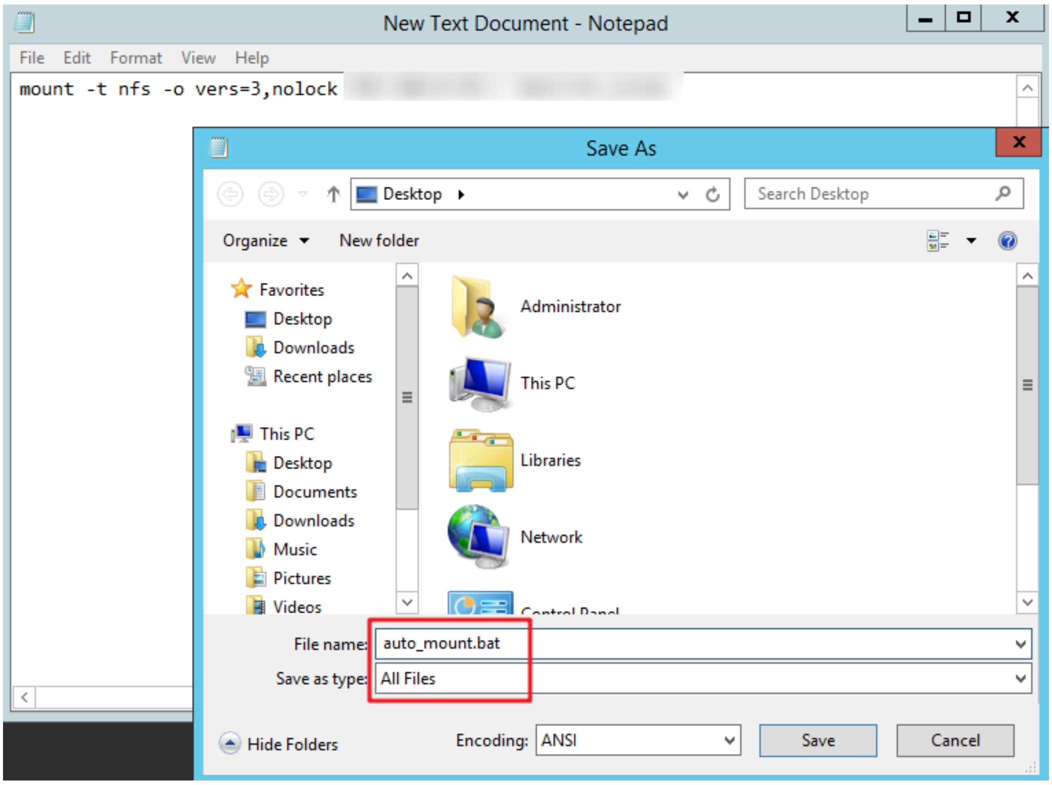
1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to switch to the ECS console.

Step 2 Log in to the ECS.

Step 3 Before mounting the file system, create a script named **auto_mount.bat**, save the script to a local host, and record the save path. The script contains the following content:

```
mount -o nolock mount point corresponding drive letter
```

Figure 2-19 Saving the script



```
mount -t nfs -o vers=3,nolock [redacted]
```

File name: auto_mount.bat

Save as type: All Files

Encoding: ANSI

For example, the **auto_mount.bat** script of a file system contains the following content:

For SFS Capacity-Oriented file systems: **mount -o nolock** *mount point* **X:**

For SFS Turbo file systems: **mount -o nolock -o casesensitive=yes** *IP address*:**/! X:**

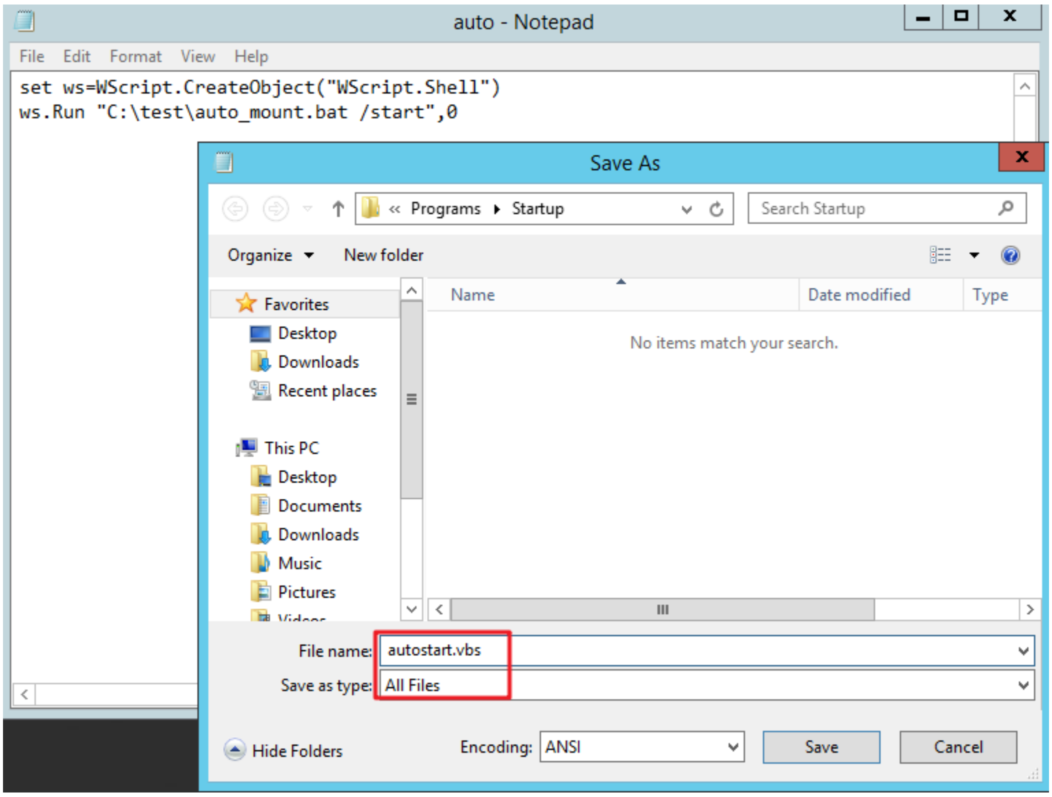
NOTE

- You can copy the mount command of the file system from the console.
- After the script is created, manually run the script in the Command Prompt to ensure that the script can be executed successfully. If you can view the file system in **This PC** after the script execution, the script can be executed properly.
- This .bat script cannot be stored in the same path in **Step 4** that stores the .vbs file. In this example, the .bat script is stored in **C:\test\**.

Step 4 Create a .txt file whose name is *XXX***.vbs** and save the file to the directory **C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup**. The file contains the following content:

```
set ws=WScript.CreateObject("WScript.Shell")
ws.Run "Local path and script name of the auto_mount.bat script /start", 0
```

Figure 2-20 Creating .vbs file



```
set ws=WScript.CreateObject("WScript.Shell")
ws.Run "C:\test\auto_mount.bat /start",0
```

File name: autostart.vbs

Save as type: All Files

Encoding: ANSI

NOTE

In this example, the local path of the **auto_mount.bat** script is **C:\test\**. Therefore, the content in the .vbs file is as follows:

```
set ws=WScript.CreateObject("WScript.Shell")
ws.Run "C:\test\auto_mount.bat /start",0
```

Step 5 After the task is created, you can restart the ECS and check whether the configuration is successful. After the configuration is successful, the file system automatically appears in **This PC**.

----End

2.4 Unmount a File System

If a file system is no longer used and needs to be deleted, you are advised to unmount the file system and then delete it.

Prerequisites

Before unmounting a file system, stop the process and read/write operations.

Linux OS

Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to go to the ECS console.

Step 2 Log in to the ECS.

Step 3 Run the following command:

umount *Local path*

Local path: An ECS local directory where the file system is mounted, for example, **/local_path**.

NOTE

Before running the **umount** command, stop all read and write operations related to the file system and exit from the local path. Or, the unmounting will fail.

----End

Windows OS

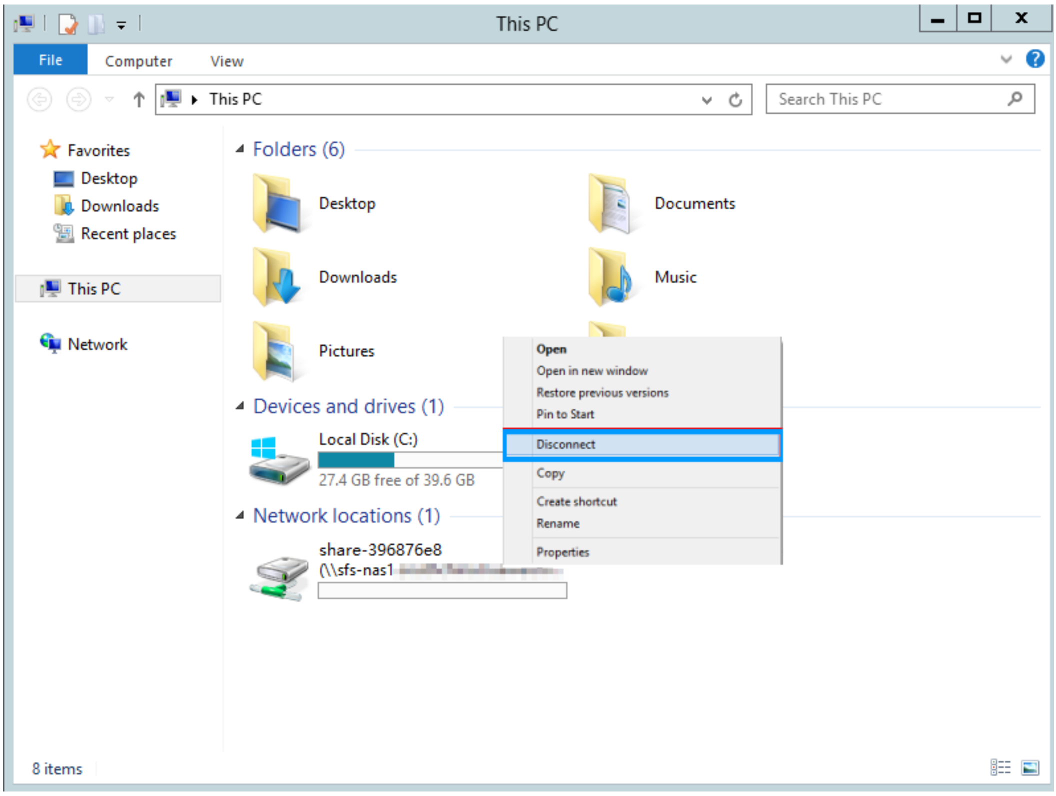
Step 1 Log in to the management console using a cloud account.

1. Log in to the management console and select a region and a project.
2. Under **Computing**, click **Elastic Cloud Server** to go to the ECS console.

Step 2 Log in to the ECS.

Step 3 Right-click the file system to be unmounted and choose **Disconnect**.

Figure 2-21 Unmounting



Step 4 If the file system disappears from the network location, it has been unmounted.

----End

3 Management

3.1 Permissions Management

3.1.1 Creating a User and Granting SFS Permissions

This chapter describes how to use IAM to implement fine-grained permissions control for your SFS 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 SFS resources.
- Grant only the permissions required for users to perform a specific task.

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

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

Prerequisites

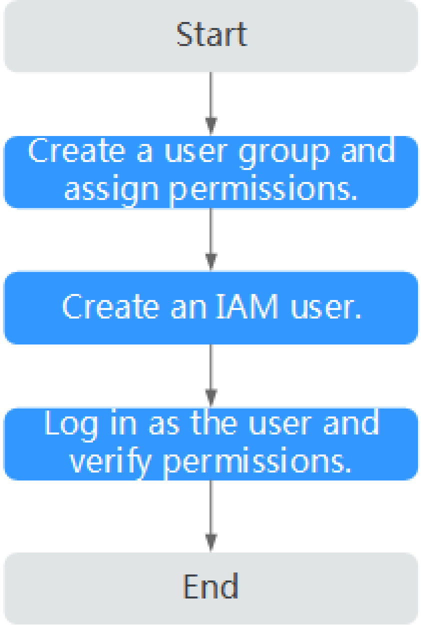
Learn about the permissions (see **Permissions**) supported by SFS and choose policies or roles according to your requirements.

Restrictions

- All system-defined policies and custom policies are supported in SFS Capacity-Oriented file systems.
- Both system-defined policies and custom policies are supported in SFS Turbo file systems.

Process Flow

Figure 3-1 Process for granting SFS permissions


Start
Create a user group and assign permissions.
Create an IAM user.
Log in as the user and verify permissions.
End

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

   Create a user group on the IAM console, and attach the **SFS ReadOnlyAccess** or **SFS Turbo 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 SFS Console using the created user, and verify that the user only has read permissions for SFS.

   - Choose **Scalable File Service**. Click **Create File System** on SFS Console. If a message appears indicating that you have insufficient permissions to perform the operation, the **SFS ReadOnlyAccess** or **SFS Turbo ReadOnlyAccess** policy has already taken effect.
   - Choose any other service. If a message appears indicating that you have insufficient permissions to access the service, the **SFS ReadOnlyAccess** or **SFS Turbo ReadOnlyAccess** policy has already taken effect.

3.1.2 Creating a Custom Policy

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

You can create custom policies in either of the following two 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.

This section provides examples of common custom SFS policies.

Example Custom Policies

- Example 1: Allowing users to create file systems

  ```
  {
          "Version": "1.1",
          "Statement": [
                  {
                          "Action": [
                                  "sfs:shares:createShare"
                          ],
                          "Effect": "Allow"
                  }
          ]
  }
  ```

- Example 2: Denying file system 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 **SFS FullAccess** policy to a user but also forbid the user from deleting file systems. Create a custom policy for denying file system deletion, and attach both policies to the group to which the user belongs. Then, the user can perform all operations on SFS except deleting file systems. The following is an example of a deny policy:

  ```
  {
          "Version": "1.1",
          "Statement": [
                  {
                          "Effect": "Deny",
                          "Action": [
                                  "sfs:shares:deleteShare"
                          ]
                  }
          ]
  }
  ```

- Example 3: Defining permissions for multiple services in a policy

  A custom policy can contain actions of multiple services that are all of the global or project-level type. The following is an example policy containing actions of multiple services:

  ```
  {
     "Version": "1.1",
     "Statement": [
        {
           "Effect": "Allow",
           "Action": [
              "sfs:shares:createShare",
              "sfs:shares:deleteShare",
              "sfs:shares:updateShare"
           ]
        },
        {
           "Effect": "Allow",
           "Action": [
              "ecs:servers:delete"
           ]
        }
     ]
  }
  ```

3.2 File System Management

3.2.1 Viewing a File System

You can search for file systems by file system name keyword or file system status, and view their basic information.

Procedure

Step 1 Log in to SFS Console.

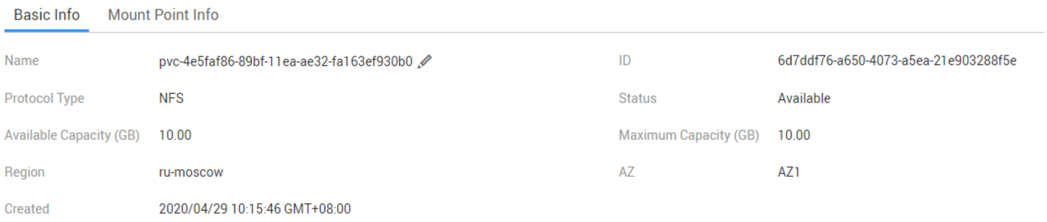
Step 2 In the file system list, view the file systems you have created. **Table 3-1** describes the file system parameters.

Table 3-1 Parameter description

Parameter	Description
Name	Name of the file system, for example, **sfs-name-001**
AZ	Availability zone where the file system is located
Status	Possible values are **Available**, **Unavailable**, **Frozen**, **Creating**, **Deleting**, and **Deletion error**.
Type	File system type
Protocol Type	File system protocol, which can be **NFS** or **CIFS**
Used Capacity (GB)	File system space already used for data storage **NOTE** This information is refreshed every 15 minutes.
Maximum Capacity (GB)	Maximum capacity of the file system
Mount Point	File system mount point. The format of an NFS file system is *File system domain name*:/*Path* or *File system IP address*:/. The format of a CIFS file system is \\*File system domain name**Path*. **NOTE** If the mount point is too long to display completely, adjust the column width.
Operation	For an SFS Capacity-Oriented file system, operations include resizing, deletion, and monitoring metric viewing. For an SFS Turbo file system, valid operations include capacity expansion and deletion.

Step 3 Click the name of a file system to view detailed information about the file system. See **Figure 3-2**.

Figure 3-2 File system information



Name	pvc-4e5faf86-89bf-11ea-ae32-fa163ef930b0	ID	6d7ddf76-a650-4073-a5ea-21e903288f5e
Protocol Type	NFS	Status	Available
Available Capacity (GB)	10.00	Maximum Capacity (GB)	10.00
Region	ru-moscow	AZ	AZ1
Created	2020/04/29 10:15:46 GMT+08:00		

Step 4 (Optional) Search for file systems by file system name keyword or file system status.

----End

3.2.2 Deleting a File System

Data in a deleted file system cannot be restored. Ensure that files in a file system have been properly stored or backed up before you delete the file system.

Prerequisites

The file system to be deleted has been unmounted. For details about how to unmount the file system, see **Unmount a File System**.

Procedure

Step 1 Log in to SFS Console.

Step 2 In the file system list, locate the file system you want to delete and click **Delete** in the **Operation** column.

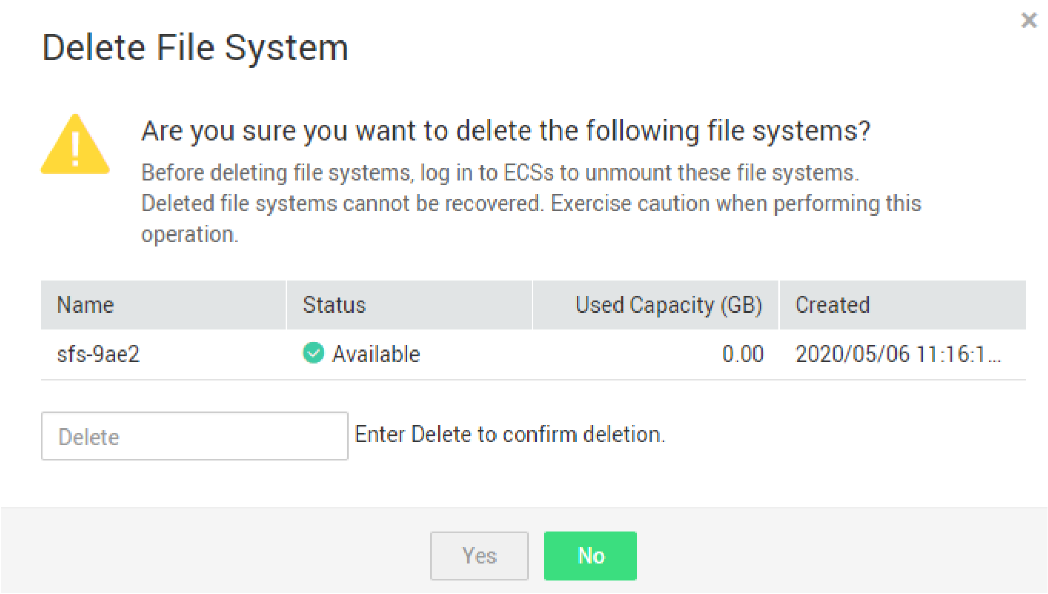
If you want to delete more than one file system at a time, select the file systems, and then click **Delete** in the upper left part of the file system list. In the displayed dialog box, confirm the information, enter **Delete** in the text box, and then click **Yes**. Batch deletion is only supported for SFS Capacity-Oriented file systems.

Step 3 In the displayed dialog box, confirm the information, enter **Delete** in the text box, and then click **Yes**.

NOTE

Only **Available** and **Unavailable** file systems can be deleted.

Figure 3-3 Deleting a file system



Delete File System

Are you sure you want to delete the following file systems?

Before deleting file systems, log in to ECSs to unmount these file systems. Deleted file systems cannot be recovered. Exercise caution when performing this operation.

Name	Status	Used Capacity (GB)	Created
sfs-9ae2	✓ Available	0.00	2020/05/06 11:16:1...

Delete Enter Delete to confirm deletion.

Step 4 Check that the file system disappears from the file system list.

----End

3.3 Network Configuration

3.3.1 Configuring Multi-VPC Access

VPC provisions an isolated virtual network environment defined and managed by yourself, improving the security of cloud resources and simplifying network deployment. When using SFS, a file system and the associated ECSs need to belong to the same VPC for file sharing.

In addition, VPC can use network access control lists (ACLs) to implement access control. A network ACL is an access control policy system for one or more subnets. Based on inbound and outbound rules, it determines whether data packets are allowed in or out of any associated subnet. In the VPC list of a file system, each time an authorized address is added and corresponding permissions are set, a network ACL is created.

For more information about VPC, see the *Virtual Private Cloud User Guide*.

Scenarios

Multi-VPC access can be configured for an SFS Capacity-Oriented file system so that ECSs in different VPCs can share the same file system, as long as the VPCs that the ECSs belong to are added to the VPC list of the file system or the ECS IP addresses are added as authorized IP addresses of the VPCs.

This section describes how to configure multi-VPC access for an SFS Capacity-Oriented file system.

Restrictions

- You can add a maximum of 20 VPCs for each file system. A maximum of 400 ACL rules for added VPCs can be created. When adding a VPC, the default IP address 0.0.0.0/0 is automatically added.
- If a VPC added to a file system has been deleted from the VPC console, the IP address/address segment of this VPC can still be seen as activated in the file system's VPC list. But this VPC can no longer be used and you are advised to delete it from the list.

Procedure

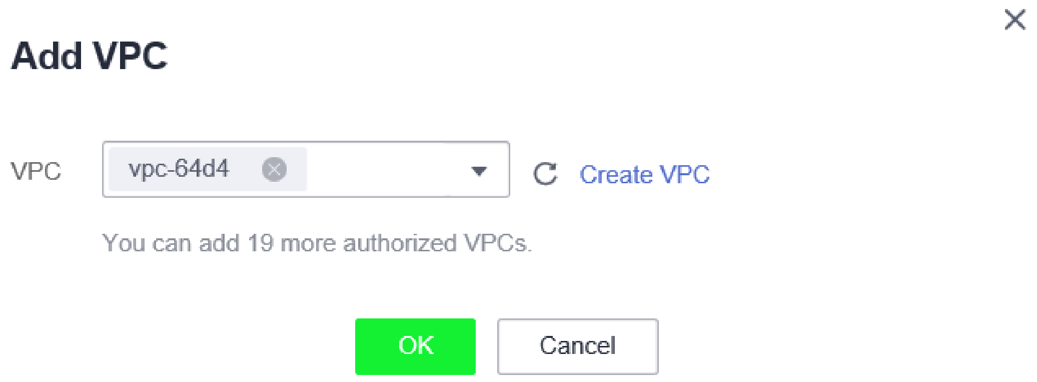
Step 1 Log in to SFS Console.

Step 2 In the file system list, click the name of the target file system. On the displayed page, locate the **Authorizations** area.

Step 3 If no VPCs are available, create one. You can add multiple VPCs for a file system. Click **Add Authorized VPC** and the **Add Authorized VPC** dialog box is displayed. See **Figure 3-4**.

You can select multiple VPCs from the drop-down list.

Figure 3-4 Adding VPCs



Add VPC

VPC: vpc-64d4

You can add 19 more authorized VPCs.

Step 4 Click **OK**. A successfully added VPC is displayed in the list. When adding a VPC, the default IP address **0.0.0.0/0** is automatically added. The default read/write permission is **Read-write**, the default user permission is **no_all_squash**, and the default root permission is **no_root_squash**.

Step 5 View the VPC information in the VPC list. For details about the parameters, see **Table 3-2**.

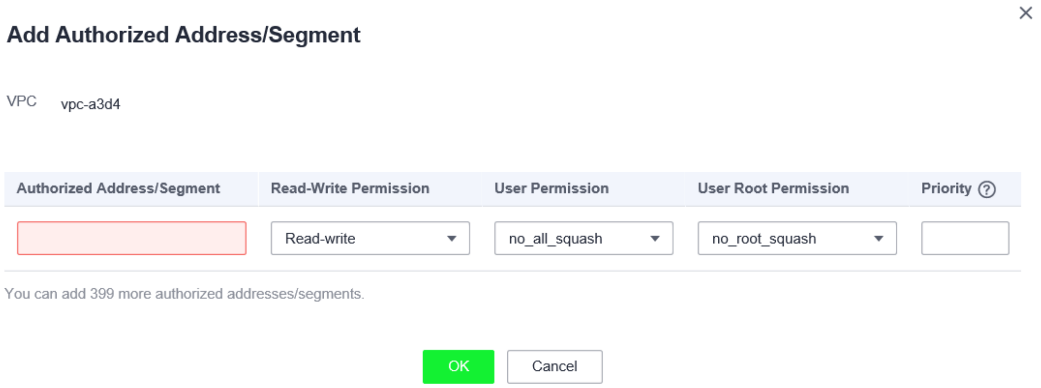
Table 3-2 Parameter description

Parameter	Description
Name	Name of the added VPC, for example, **vpc-01**
Authorized Addresses/Segments	Number of added IP addresses or IP address segments

Parameter	Description
Operation	The value can be **Add** or **Delete**. **Add**: Adds an authorized VPC. This operation configures the IP address, read/write permission, user permission, user root permission, and priority. For details, see **Table 3-3**. **Delete**: Deletes this VPC.

Step 6 Click ⌄ on the left of the VPC name to view details about the IP addresses/segments added to this VPC. You can add, edit, or delete IP addresses/segments. In the **Operation** column of the target VPC, click **Add**. The **Add Authorized Address/Segment** dialog box is displayed. See **Figure 3-5**. **Table 3-3** describes the parameters to be configured.

Figure 3-5 Adding an authorized address or segment



Add Authorized Address/Segment

VPC vpc-a3d4

Authorized Address/Segment	Read-Write Permission	User Permission	User Root Permission	Priority
	Read-write	no_all_squash	no_root_squash	

You can add 399 more authorized addresses/segments.

Table 3-3 Parameter description

Parameter	Description
Authorized Address/Segment	● Enter one IPv4 address or address segment at a time. ● The entered IPv4 address or address segment must be valid and cannot be one starting with 0 except 0.0.0.0/0. If you add **0.0.0.0/0**, any IP address within this VPC will be authorized for accessing the file system. Class D and class E IP addresses are not supported. Therefore, do not enter an IP address or address segment starting with any number ranging from 224 to 255, for example 224.0.0.1 or 255.255.255.255. IP addresses or address segments starting with 127 are also not supported. If an invalid IP address or address segment is used, the access rule may fail to be added or the added access rule cannot take effect. ● Do not enter multiple IP addresses (separated using commas) at a time. For example, do not enter 10.0.1.32,10.5.5.10. ● If you enter an IP address segment, enter it in the format of *IP address/mask*. For example, enter 192.168.1.0/24. Do not enter in the format of 192.168.1.0-255 or 192.168.1.0-192.168.1.255. The number of bits in a subnet mask must be an integer ranging from 0 to 31, and mask value **0** is valid only in 0.0.0.0/0.
Read-Write Permission	The value can be **Read-write** or **Read-only**. The default value is **Read-write**.
User Permission	Whether to retain the user identifier (UID) and group identifier (GID) of the shared directory. The default value is **no_all_squash**. ● **all_squash**: The UID and GID of a shared directory are mapped to user **nobody**, which is applicable to public directories. ● **no_all_squash**: The UID and GID of a shared directory are retained. This parameter is not involved when an authorized address is added for a CIFS file system.

Parameter	Description
User Root Permission	Whether to allow the root permission of the client. The default value is **no_root_squash**. ● **root_squash**: Clients cannot access as the **root** user. When a client accesses as the **root** user, the user is mapped to the **nobody** user. ● **no_root_squash**: Clients are allowed to access as the **root** user who has full control and access permissions of the root directories. This parameter is not involved when an authorized address is added for a CIFS file system.
Priority	The value must be an integer ranging from **0** to **100**. **0** indicates the highest priority, and **100** indicates the lowest priority. In the same VPC, the permission of the IP address or address segment with the highest priority is preferentially used. If some IP addresses or address segments are of the same priority, the permission of the most recently added or modified one is used. For example, if the IP address for mounting is 10.1.1.32 and both 10.1.1.32 (read/write) with priority **100** and 10.1.1.0/24 (read-only) with priority **50** meet the requirements, the permission of 10.1.1.0/24 (read-only) with priority **50** is used. That is, if there is no other authorized priority, the permission of all IP addresses in the 10.1.1.0/24 segment, including 10.1.1.32, is read-only.

NOTE

For an ECS in VPC A, its IP address can be added to the authorized IP address list of VPC B, but the file system of VPC B cannot be mounted to this ECS. The VPC of the ECS and the file system must be the same.

----End

Verification

After another VPC is configured for the file system, if the file system can be mounted to ECSs in the VPC and the ECSs can access the file system, the configuration is successful.

Example

A user creates an SFS Capacity-Oriented file system A in VPC-B. The network segment is **10.0.0.0/16**. The user has an ECS D in VPC-C, using the private IP address **192.168.10.11** in network segment **192.168.10.0/24**. If the user wants to mount file system A to ECS D and allow the file system to be read and written, the user needs to add VPC-C to file system A's VPC list, add ECS D's private IP address or address segment to the authorized addresses of VPC-C, and then set **Read-Write Permission** to **Read-write**.

The user purchases an ECS F that uses the private IP address **192.168.10.22** in the VPC-C network segment **192.168.10.0/24**. If the user wants ECS F to have only the read permission for file system A and its read priority to be lower than that of ECS D, the user needs to add ECS F's private IP address to VPC-C's authorized addresses, set **Read-Write Permission** to **Read-only**, and set **Priority** to an integer between 0 and 100 and greater than the priority set for ECS D.

3.3.2 Configuring Multi-Account Access

Scenarios

In addition to multi-VPC access, SFS Capacity-Oriented file systems also support cross-VPC access with different accounts.

If VPC IDs used by other accounts are added to the authorization list of a file system, and IP addresses or address segments of ECSs are added to the authorized address list, ECSs under different accounts can share the same file system.

For more information about VPC, see the *Virtual Private Cloud User Guide*.

Restrictions

- You can add a maximum of 20 VPCs for each file system. A maximum of 400 ACL rules for added VPCs can be created.
- If a VPC bound to the file system has been deleted from the VPC console, the IP address/address segment of this VPC in the VPC list of the file system can still be seen as activated. However, this VPC cannot be used any longer and you are advised to delete the VPC from the list.

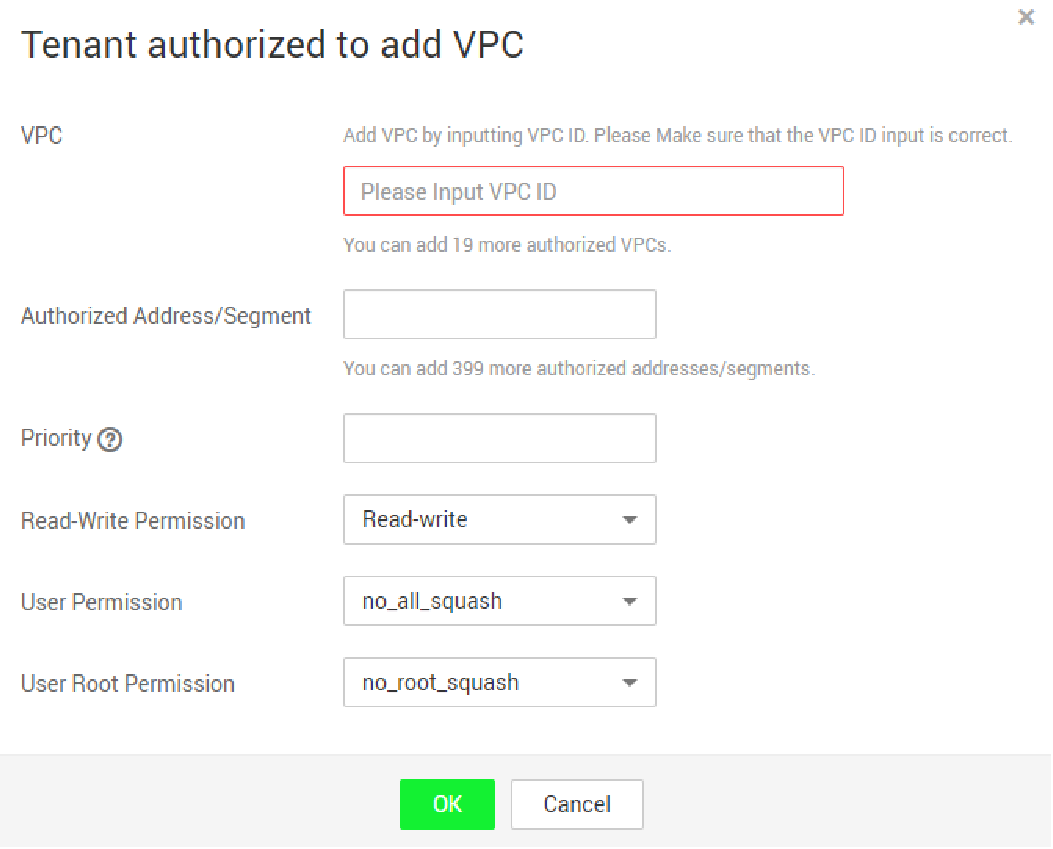
Procedure

Step 1 Log in to SFS Console.

Step 2 In the file system list, click the name of the target file system. On the displayed page, locate the **Authorizations** area.

Step 3 Click **Tenant authorized to add VPC** to add VPCs used by other accounts for the file system. A dialog box is displayed. See **Figure 3-6**.

Figure 3-6 Adding a VPC of an authorized tenant



Tenant authorized to add VPC

VPC: Add VPC by inputting VPC ID. Please Make sure that the VPC ID input is correct.

Please Input VPC ID

You can add 19 more authorized VPCs.

Authorized Address/Segment:

You can add 399 more authorized addresses/segments.

Priority:

Read-Write Permission: Read-write

User Permission: no_all_squash

User Root Permission: no_root_squash

Table 3-4 describes the parameters to be configured.

Table 3-4 Parameter description

Parameter	Description
VPC	Enter the VPC ID of the VPC to be added. You can obtain the VPC ID on the details page of the target VPC on the VPC console.

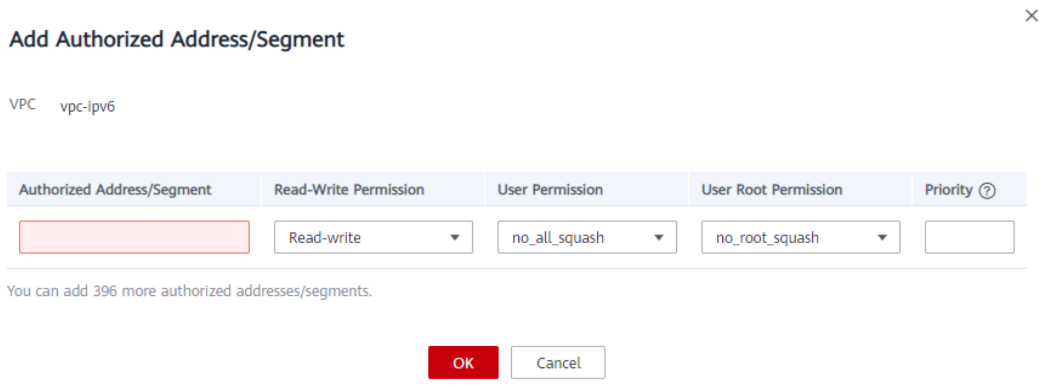
Parameter	Description
Authorized Address/Segment	● Only one IPv4 address or address segment can be entered. ● The entered IPv4 address or address segment must be valid and cannot be an IP address or address segment starting with 0 except 0.0.0.0/0. The value **0.0.0.0/0** indicates any IP address in the VPC. In addition, the IP address or address segment cannot start with 127 or any number from 224 to 255, such as 127.0.0.1, 224.0.0.1, or 255.255.255.255. This is because IP addresses or address segments starting with any number from 224 to 239 are class D addresses and they are reserved for multicast. IP addresses or address segments starting with any number from 240 to 255 are class E addresses and they are reserved for research purposes. If an invalid IP address or address segment is used, the access rule may fail to be added or the added access rule cannot take effect. ● Multiple addresses separated by commas (,), such as **10.0.1.32,10.5.5.10** are not allowed. ● An address segment, for example, 192.168.1.0 to 192.168.1.255, needs to be in the mask format like 192.168.1.0/24. Other formats such as 192.168.1.0-255 are not allowed. The number of bits in a subnet mask must be an integer ranging from 0 to 31. The number of bits **0** is valid only in 0.0.0.0/0.
Priority	The value must be an integer ranging from **0** to **100**. **0** indicates the highest priority, and **100** indicates the lowest priority. In the same VPC, the permission of the IP address or address segment with the highest priority is preferentially used. If some IP addresses or address segments are of the same priority, the permission of the most recently added or modified one is used. For example, if the IP address for mounting is 10.1.1.32 and both 10.1.1.32 (read/write) with priority **100** and 10.1.1.0/24 (read-only) with priority **50** meet the requirements, the permission of 10.1.1.0/24 (read-only) with priority **50** is used. That is, if there is no other authorized priority, the permission of all IP addresses in the 10.1.1.0/24 segment, including 10.1.1.32, is read-only.
Read&Write Permissions	The value can be **Read&Write** or **Read-only**. The default value is **Read&Write**.

Parameter	Description
User Permission	Specifies whether to retain the user identifier (UID) and group identifier (GID) of the shared directory. The default value is **no_all_squash**. ● **all_squash**: The UID and GID of a shared directory are mapped to user **nobody**, which is applicable to public directories. ● **no_all_squash**: The UID and GID of a shared directory are retained. This parameter is not involved when an authorized address is added for a CIFS file system.
User Root Permission	Specifies whether to allow the root permission of the client. The default value is **no_root_squash**. ● **root_squash**: Clients cannot access as the **root** user. When a client accesses as the **root** user, the user is mapped to the **nobody** user. ● **no_root_squash**: Clients are allowed to access as the **root** user who has full control and access permissions of the root directories. This parameter is not involved when an authorized address is added for a CIFS file system.

Step 4 Click **OK**. The added VPC is displayed in the list.

Step 5 Click ∨ on the left of the VPC name to view details about the IP addresses/segments added to this VPC. You can add, edit, or delete IP addresses/segments. In the **Operation** column of the target VPC, click **Add**. The **Add Authorized Address/Segment** dialog box is displayed. See **Figure 3-7**. **Table 3-4** describes the parameters to be added.

Figure 3-7 Adding an authorized address or segment



Add Authorized Address/Segment

VPC vpc-ipv6

Authorized Address/Segment	Read-Write Permission	User Permission	User Root Permission	Priority ⓘ
	Read-write ▾	no_all_squash ▾	no_root_squash ▾	

You can add 396 more authorized addresses/segments.

----End

Verification

After another user's VPC is configured for the file system, if the file system can be mounted to ECSs in the VPC and the ECSs can access the file system, the configuration is successful.

3.3.3 Configuring DNS

A DNS server is used to resolve domain names of file systems.

Scenarios

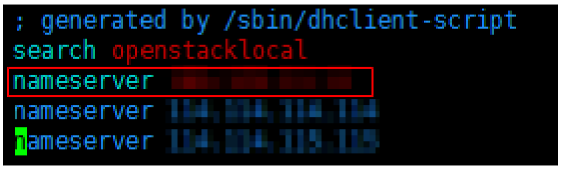
By default, the IP address of the DNS server used to resolve domain names of file systems is automatically configured on ECSs when creating ECSs. No manual configuration is needed except when the resolution fails due to a change in the DNS server IP address.

Procedure

Step 1 Log in to the ECS as user **root**.

Step 2 Run the **vi /etc/resolv.conf** command to edit the **/etc/resolv.conf** file. Add the DNS server IP address above the existing nameserver information. See **Figure 3-8**.

Figure 3-8 Configuring DNS



```
; generated by /sbin/dhclient-script
search openstacklocal
nameserver [redacted]
nameserver [redacted]
nameserver [redacted]
```

The format is as follows:

```
nameserver DNS server IP address
```

Step 3 Press **Esc**, input **:wq**, and press **Enter** to save the changes and exit the vi editor.

Step 4 Run the following command to check whether the IP address is successfully added:

cat /etc/resolv.conf

Step 5 Run the following command to check whether an IP address can be resolved from the file system domain name:

nslookup *File system domain name*

NOTE

Obtain the file system domain name from the file system mount point.

Step 6 (Optional) In a network environment of the DHCP server, edit the **/etc/resolv.conf** file to prevent the file from being automatically modified upon an ECS startup, and prevent the DNS server IP address added in **Step 2** from being reset.

1. Run the following command to lock the file:

   chattr +i /etc/resolv.conf

NOTE

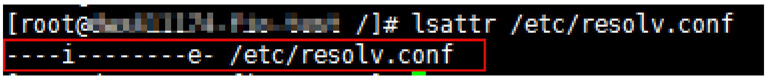
Run the **chattr -i /etc/resolv.conf** command to unlock the file if needed.

2. Run the following command to check whether the editing is successful:

   lsattr /etc/resolv.conf

   If the information shown in **Figure 3-9** is displayed, the file is locked.

   Figure 3-9 A locked file



```
[root@[redacted] /]# lsattr /etc/resolv.conf
----i--------e- /etc/resolv.conf
```

----End

3.4 File System Resizing

Scenarios

You can expand or shrink the capacity of a file system when needed.

Constraints

SFS Turbo file systems can only have their capacities expanded, not reduced. And only **In-use** file systems can be expanded.

SFS Capacity-Oriented file systems support resizing if auto capacity expansion is disabled. You can only enable auto capacity expansion when creating a file system. Once enabled, auto capacity expansion cannot be disabled, and you cannot reconfigure a maximum capacity.

SFS Capacity-Oriented file systems support resizing, during which services are not affected. Only **In-use** file systems can be expanded.

Rules for Resizing

The rules for resizing an SFS Capacity-Oriented file system are as follows:

- Expanding a file system

  Total capacity of a file system after expansion ≤ (Capacity quota of the cloud account - Total capacity of all the other file systems owned by the cloud account)

  For example, cloud account A has a quota of 500 TB. This account has already created three file systems: SFS1 (350 TB), SFS2 (50 TB), and SFS3 (70 TB). If this account needs to expand SFS2, the new capacity of SFS2 cannot be greater than 80 TB. Otherwise, the system will display a message indicating an insufficient quota and the expansion operation will fail.
- Shrinking a file system
  - When a shrink error or failure occurs on a file system, it takes approximately five minutes for the file system to restore to the available state.

– After a shrink operation fails, you can only reattempt to shrink the file system storage capacity but cannot expand it directly.

– Total capacity of a file system after shrinking ≥ Used capacity of the file system

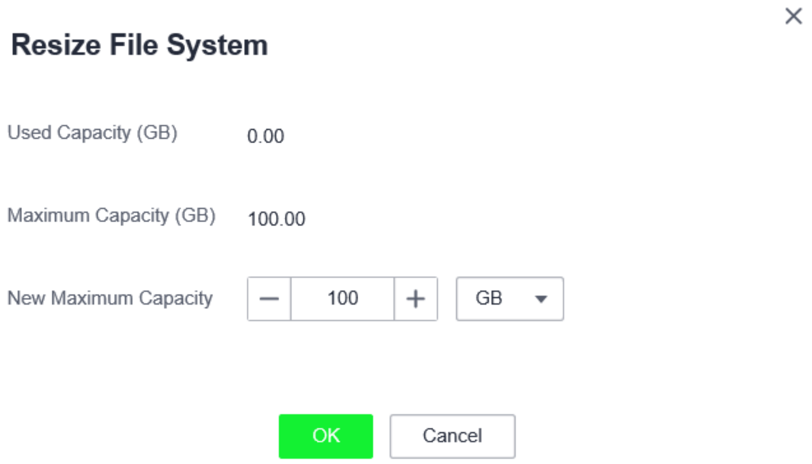
  For example, cloud account B has created a file system, SFS1. The total capacity and used capacity of SFS1 are 50 TB and 10 TB respectively. When shrinking SFS1, the user cannot set the new capacity to be smaller than 10 TB.

Procedure

Step 1 Log in to SFS Console.

Step 2 In the file system list, click **Resize** or **Expand Capacity** in the row of the desired file system. The following dialog box is displayed. See **Figure 3-10**.

Figure 3-10 Resizing a file system



Resize File System

Used Capacity (GB) 0.00

Maximum Capacity (GB) 100.00

New Maximum Capacity 100 GB

Step 3 Enter a new maximum capacity of the file system based on service requirements, and click **OK**. **Table 3-5** describes the parameters.

Table 3-5 Parameter description

Parameter	Description
Used Capacity (GB)	Used capacity of the current file system
Maximum Capacity (GB)	Maximum capacity of the current file system
New Maximum Capacity (GB)	Target maximum capacity of the file system after expanding or shrinking The value ranges from **1 GB** to **512,000 GB**. **NOTE** The new maximum capacity cannot be smaller than the used capacity.

Step 4 In the displayed dialog box, confirm the information and click **OK**.

Step 5 In the file system list, check the capacity information after resizing.

----End

3.5 Quotas

What Is Quota?

Quotas can limit the number or amount of resources available to users, such as the maximum number of ECSs or EVS disks that can be created.

If the existing resource quota cannot meet your service requirements, you can apply for a higher quota.

How Do I View My Quotas?

1. Log in to the management console.
2. Click 📍 in the upper left corner and select the desired region and project.
3. In the upper right corner of the page, click 📊 .

   The **Service Quota** page is displayed.
4. 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.

How Do I Apply for a Higher Quota?

The system does not support online quota adjustment. If you need to adjust a quota, call the hotline or send an email to the customer service mailbox. Customer service personnel will timely process your request for quota adjustment and inform you of the real-time progress by making a call or sending an email.

Before dialing the hotline number or sending an email, make sure that the following information has been obtained:

- Account name, project name, and project ID, which can be obtained by performing the following operations:

  Log in to the management console using the cloud account, click the username in the upper right corner, select **My Credentials** from the drop-down list, and obtain the account name, project name, and project ID on the **My Credentials** page.
- Quota information, which includes:
  - Service name
  - Quota type
  - Required quota

Learn how to obtain the service hotline and email address.

3.6 Encryption

Creating an Encrypted File System

To use the file system encryption function, you can directly select the encryption function when creating an SFS Turbo file system without authorization. For details, see **File System Encryption**.

You can create a file system that is encrypted or not, but you cannot change the encryption settings of an existing file system.

For details about how to create an encrypted file system, see **Create a File System**.

Unmounting an Encrypted File System

If the custom key used by the encrypted file system is disabled or scheduled for deletion, the file system can only be used within a certain period of time (30s by default). Exercise caution in this case.

For details about how to unmount the file system, see **Unmount a File System**.

3.7 Backup

Only SFS Turbo file systems can be backed up using CBR while SFS Capacity-Oriented file systems cannot.

Scenarios

A backup is a complete copy of an SFS Turbo file system at a specific time and it records all configuration data and service data at that time.

For example, if a file system is faulty or encounters a logical error (for example, mis-deletion, hacker attacks, and virus infection), you can use data backups to restore data quickly.

Creating a File System Backup

Ensure that the target file system is available. Or, the backup task cannot start. This procedure describes how to manually create a file system backup.

NOTE

If any modification is made to a file system during the backup, inconsistencies may occur. For example, there may be duplicate or deleted data, or data discrepancies. Such a modification includes a write, rename, move or delete. To ensure backup data consistency, you are advised to stop the applications or programs that use the file system during the backup, or schedule the backup at off-peak hours.

Step 1 Log in to CBR Console.

Step 2 In the navigation pane on the left, choose **SFS Turbo Backups**.

Step 3 Create a backup vault by following the instructions in section "Creating an SFS Turbo Backup Vault" in the *Cloud Backup and Recovery User Guide*. Then, create a backup by following the instructions in section "Creating an SFS Turbo Backup."

Step 4 The system automatically backs up the file system.

You can view the backup creation status on the **Backups** tab page. When the **Status** of the backup changes to **Available**, the backup has been created.

Step 5 If the file system becomes faulty or an error occurred, you can restore the backup data to a new file system. For details, see **Using a Backup to Create a File System**.

----End

Using a Backup to Create a File System

In case of a virus attack, accidental deletion, or software or hardware fault, you can use an SFS Turbo file system backup to create a new file system. Data on the new file system is the same as that in the backup.

Step 1 Log in to CBR Console.

1. Log in to the management console.
2. Click 📍 in the upper left corner and select your desired region and project.
3. Choose **Storage** > **Cloud Backup and Recovery** > **SFS Turbo Backups**.

Step 2 Click the **Backups** tab and locate the desired backup.

Step 3 If the status of the target backup is **Available**, click **Create File System** in the **Operation** column of the backup.

NOTE

For how to create backups, see sections "Purchasing an SFS Turbo Backup Vault" and "Creating an SFS Turbo Backup".

Step 4 Set the file system parameters.

NOTE

- For detailed parameter descriptions, see table "Parameter description" under **Creating an SFS Turbo File System**.

Step 5 Click **Next**.

Step 6 Go back to the file system list and check whether the file system is successfully created.

You will see the file system status change as follows: **Creating**, **Available**, **Restoring**, **Available**. You may not notice the **Restoring** status because Instant Restore is supported and the restoration speed is very fast. After the file system status has changed from **Creating** to **Available**, the file system is successfully created. After the status has changed from **Restoring** to **Available**, backup data has been successfully restored to the created file system.

----End

3.8 Monitoring

3.8.1 SFS Metrics

Function

This section describes metrics reported by Scalable File Service (SFS) as well as their namespaces and dimensions. You can use the console or APIs provided by Cloud Eye to query the metrics generated for SFS.

Namespace

SYS.SFS

Metrics

Table 3-6 SFS metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
read_bandwidth	Read Bandwidth	Read bandwidth of a file system within a monitoring period Unit: byte/s	≥ 0 bytes/s	SFS file system	4 minutes
write_bandwidth	Write Bandwidth	Write bandwidth of a file system within a monitoring period Unit: byte/s	≥ 0 bytes/s	SFS file system	4 minutes
rw_bandwidth	Read and Write Bandwidth	Read and write bandwidth of a file system within a monitoring period Unit: byte/s	≥ 0 bytes/s	SFS file system	4 minutes

Dimension

Key	Value
share_id	SFS file system

Viewing Monitoring Statistics

Step 1 Log in to the management console.

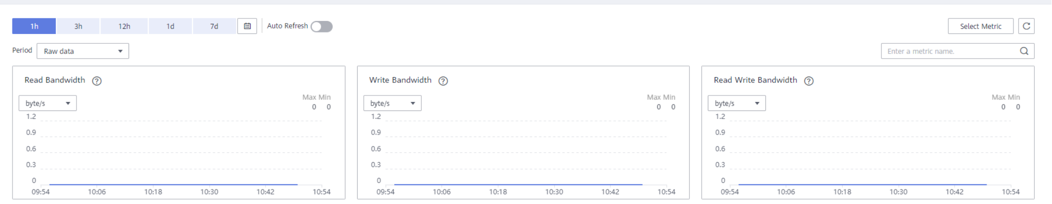
Step 2 View the monitoring graphs using either of the following methods.

- Method 1: Choose **Service List** > **Storage** > **Scalable File Service**. In the file system list, click **View Metric** in the **Operation** column of the target file system.
- Method 2: Choose **Management & Deployment** > **Cloud Eye** > **Cloud Service Monitoring** > **Scalable File Service**. In the file system list, click **View Metric** in the **Operation** column of the target file system.

Step 3 View the SFS file system monitoring data by metric or monitored duration.

Figure 3-11 shows the monitoring graphs. For more information about Cloud Eye, see the *Cloud Eye User Guide*.

Figure 3-11 SFS monitoring graphs



Period Raw data

----End

3.9 Auditing

Scenarios

Cloud Trace Service (CTS) records operations of SFS resources, facilitating query, audit, and backtracking.

Prerequisites

You have enabled CTS and the tracker is normal. For details about how to enable CTS, see section "Enabling CTS" in the *Cloud Trace Service User Guide.*

Operations

Table 3-7 SFS operations traced by CTS

Operation	Resource Type	Trace
Creating a shared file system	sfs	createShare
Modifying a shared file system	sfs	updateShareInfo

Operation	Resource Type	Trace
Deleting a shared file system	sfs	deleteShare
Adding a share access rule	sfs	addShareACL
Deleting a share access rule	sfs	deleteShareACL

Table 3-8 SFS Turbo operations traced by CTS

Operation	Resource Type	Trace
Creating a file system	sfs_turbo	createShare
Deleting a file system	sfs_turbo	deleteShare

Querying Traces

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner and select a region and project.

Step 3 Choose **Management & Deployment** > **Cloud Trace Service**.

The **Cloud Trace Service** page is displayed.

Step 4 In the navigation pane on the left, choose **Trace List**.

Step 5 On the trace list page, set **Trace Source**, **Resource Type**, and **Search By**, and click **Query** to query the specified traces.

For details about other operations, see section "Querying Real-Time Traces" in the *Cloud Trace Service User Guide*.

----End

Disabling or Enabling a Tracker

This section describes how to disable an existing tracker on the CTS console. After the tracker is disabled, the system will stop recording operations, but you can still view existing operation records.

Step 1 Log in to the management console.

Step 2 Choose **Management & Governance** > **Cloud Trace Service**.

The **Cloud Trace Service** page is displayed.

Step 3 Click **Trackers** in the left pane.

Step 4 Click **Disable** on the right of the tracker information.

Step 5 Click **Yes**.

Step 6 After the tracker is disabled, the available operation changes from **Disable** to **Enable**. To enable the tracker again, click **Enable** and then click **Yes**. The system will start recording operations again.

----End

4 Typical Applications

4.1 HPC

Context

HPC is short for high-performance computing. An HPC system or environment is made up of a single computer system with many CPUs, or a cluster of multiple computer clusters. It can handle a large amount of data and perform high-performance computing that would be rather difficult for PCs. HPC has ultra-high capability in floating-point computation and can be used for compute-intensive and data-intensive fields, such as industrial design, bioscience, energy exploration, image rendering, and heterogeneous computing. Different scenarios put different requirements on the file system:

- Industrial design: In automobile manufacturing, CAE and CAD simulation software are widely used. When the software is operating, compute nodes need to communicate with each other closely, which requires high bandwidth and low latency of the file system.
- Bioscience: The file system should have high bandwidth and large storage, and be easy to expand.
  - Bioinformatics: To sequence, stitch, and compare genes.
  - Molecular dynamics: To simulate the changes of proteins at molecular and atomic levels.
  - New drug R&D: To complete high-throughput screening (HTS) to shorten the R&D cycle and reduce the investment.
- Energy exploration: Field operations, geologic prospecting, geological data processing and interpretation, and identification of oil and gas reservoirs all require large memory and high bandwidth of the file system.
- Image rendering: Image processing, 3D rendering, and frequent processing of small files require high read/write performance, large capacity, and high bandwidth of file systems.
- Heterogeneous computing: Compute elements may have different instruction set architectures, requiring the file system provide high bandwidth and low latency.

SFS is a shared storage service based on file systems. It features high-speed data sharing, dynamic storage tiering, as well as on-demand, smooth, and online resizing. These outstanding features empower SFS to meet the demanding requirements of HPC on storage capacity, throughput, IOPS, and latency.

A biological company needs to perform plenty of gene sequencing using software. However, due to the trivial steps, slow deployment, complex process, and low efficiency, self-built clusters are reluctant to keep abreast of business development. However, things are getting better since the company resorted to professional HPC service process management software. With massive compute and storage resource of the cloud platform, the initial investment and cost during O&M are greatly reduced, the service rollout time is shortened, and efficiency is boosted.

Configuration Process

1. Organize the files of DNA sequencing to be uploaded.
2. Log in to SFS Console. Create a file system to store the files of DNA sequencing.
3. Log in to the servers that function as the head node and compute node, and mount the file system.
4. On the head node, upload the files to the file system.
5. On the compute node, edit the files.

Prerequisites

- A VPC has been created.
- ECSs that function as head nodes and compute nodes have been created, and have been assigned to the VPC.
- SFS has been enabled.

Example Configuration

Step 1 Log in to SFS Console.

Step 2 In the navigation pane, choose **SFS Capacity-Oriented**. In the upper right corner of the page, click **Create File System**.

Step 3 On the **Create File System** page, set parameters as instructed.

Step 4 After the configuration is complete, click **Create Now**.

To mount a file system to Linux ECSs, see **Mounting an NFS File System to ECSs (Linux)**. To mount a file system to Windows ECSs, see **Mounting an NFS File System to ECSs (Windows)** and **Mounting a CIFS File System to ECSs (Windows)**.

Step 5 Log in to the head node, and upload the files to the file system.

Step 6 Start gene sequencing, and the compute node obtains the gene sequencing file from the mounted file system for calculation.

----End

4.2 Media Processing

Context

Media processing involves uploading, downloading, cataloging, transcoding, and archiving media materials, as well as storing, invoking, and managing audio and video data. Media processing has the following requirements on shared file systems:

- Media materials feature a high video bit rate and a large scale. The capacity of file systems must be large and easy to be expanded.
- Acquisition, editing, and synthesis of audio and video data require stable and low-latency file systems.
- Concurrent editing requires file systems to deliver reliable and easy-to-use data sharing.
- Video rendering and special effects need processing small files frequently. The file systems must offer high I/O performance.

SFS is a shared storage service based on file systems. It features high-speed data sharing, dynamic storage tiering, as well as on-demand, smooth, and online resizing. These outstanding features empower SFS to meet the demanding requirements of media processing on storage capacity, throughput, IOPS, and latency.

A TV channel has a large volume of audio and video materials to process. The work will be done on multiple editing workstations. The TV channel uses SFS to enable file sharing among the editing workstations. First, a file system is mounted to ECSs that function as upload workstations and editing workstations. Then raw materials are uploaded to the shared file system through the upload workstations. Then, the editing workstations concurrently edit the materials in the shared file system.

Configuration Process

1. Organize the material files that are to be uploaded.
2. Log in to SFS Console. Create a file system to store the material files.
3. Log in to the ECSs that function as upload workstations and editing workstations, and mount the file system.
4. On the upload workstations, upload the material files to the file system.
5. On the editing stations, edit the material files.

Prerequisites

- A VPC has been created.
- ECSs that function as upload workstations and editing workstations have been created, and have been assigned to the VPC.
- SFS has been enabled.

Example Configuration

Step 1 Log in to SFS Console.

Step 2 In the navigation pane, choose **SFS Capacity-Oriented**. In the upper right corner of the page, click **Create File System**.

Step 3 On the **Create File System** page, set parameters as instructed.

Step 4 After the configuration is complete, click **Create Now**.

To mount a file system to Linux ECSs, see **Mounting an NFS File System to ECSs (Linux)**. To mount a file system to Windows ECSs, see **Mounting an NFS File System to ECSs (Windows)** and **Mounting a CIFS File System to ECSs (Windows)**.

Step 5 Log in to the upload workstations, and upload the material files to the file system.

Step 6 Log in to the editing workstations, and edit the material files.

----End

4.3 Enterprise Website/App Background

Context

For I/O-intensive website services, SFS Turbo can provide shared website source code directories and storage for multiple web servers, enabling low-latency and high-IOPS concurrent share access. Features of such services are as follows:

- A large number of small files: Static website files need to be stored, including HTML files, JSON files, and static images.
- Read I/O intensive: Scope of data reading is large, and data writing is relatively small.
- Multiple web servers access an SFS Turbo background to achieve high availability of website services.

Configuration Process

1. Sort out the website files.
2. Log in to SFS Console. Create an SFS Turbo file system to store the website files.
3. Log in to the server that functions as the compute node and mount the file system.
4. On the head node, upload the files to the file system.
5. Start the web server.

Prerequisites

- A VPC has been created.
- Servers that function as head nodes and compute nodes have been created, and have been assigned to the VPC.

- SFS has been enabled.

Example Configuration

Step 1 Log in to SFS Console.

Step 2 In the navigation pane, choose **SFS Capacity-Oriented**. In the upper right corner of the page, click **Create File System**.

Step 3 On the **Create File System** page, set parameters as instructed.

Step 4 After the configuration is complete, click **Create Now**.

To mount a file system to Linux ECSs, see **Mounting an NFS File System to ECSs (Linux)**. To mount a file system to Windows ECSs, see **Mounting an NFS File System to ECSs (Windows)** and **Mounting a CIFS File System to ECSs (Windows)**.

Step 5 Log in to the head node and upload the files to the file system.

Step 6 Start the web server.

----End

4.4 Log Printing

Context

SFS Turbo can provide multiple service nodes for shared log output directories, facilitating log collection and management of distributed applications. Features of such services are as follows:

- A shared file system is mounted to multiple service hosts and logs are printed concurrently.
- Large file size and small I/O: The size of a single log file is large, but the I/O of each log writing is small.
- Write I/O intensive: Write I/O of small blocks is the major service.

Configuration Process

1. Log in to SFS Console. Create an SFS Turbo file system to store the log files.
2. Log in to the server that functions as the compute node and mount the file system.
3. Configure the log directory to the shared file system. It is recommended that each host use different log files.
4. Start applications.

Prerequisites

- A VPC has been created.
- Servers that function as head nodes and compute nodes have been created, and have been assigned to the VPC.
- SFS has been enabled.

Example Configuration

Step 1 Log in to SFS Console.

Step 2 In the navigation pane, choose **SFS Capacity-Oriented**. In the upper right corner of the page, click **Create File System**.

Step 3 On the **Create File System** page, set parameters as instructed.

Step 4 After the configuration is complete, click **Create Now**.

To mount a file system to Linux ECSs, see **Mounting an NFS File System to ECSs (Linux)**. To mount a file system to Windows ECSs, see **Mounting an NFS File System to ECSs (Windows)** and **Mounting a CIFS File System to ECSs (Windows)**.

Step 5 Configure the log directory to the shared file system. It is recommended that each host use different log files.

Step 6 Start applications.

----End

5 Troubleshooting

5.1 Mounting a File System Times Out

Symptom

When a file system is mounted to servers using the **mount** command, message **timed out** is displayed.

Possible Causes

- Cause 1: The network status is not stable.
- Cause 2: The network connection is abnormal.
- Cause 3: The DNS configuration of the server is incorrect. As a result, the domain name of the file system cannot be resolved, and the mounting fails. This issue will not occur on SFS Turbo file systems.
- Cause 4: The server where the file system is to be mounted runs Ubuntu18 or later.

Fault Diagnosis

After the network fault is excluded, run the **mount** command again.

Solution

- Cause 1 and Cause 2: The network status is not stable or the network connection is abnormal.

  Re-mount the file system after the network issue is addressed.

  - If the patch is uninstalled successfully, no further action is required.
  - If the problem persists, see the solution for cause 3.
- Cause 3: The DNS configuration of the server is incorrect. As a result, the domain name of the file system cannot be resolved, and the mounting fails.

  a. Check the DNS configuration of the tenant and run the **cat /etc/resolv.conf** command.

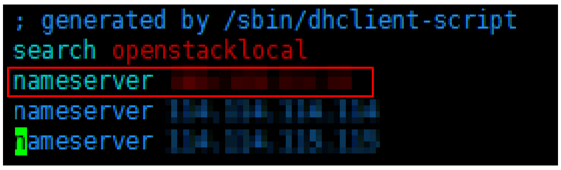
- If the DNS has not been configured, configure it. For details about how to configure the DNS, see **Configuring DNS**.
- If the DNS has been configured, run the following command to check whether the DNS is correct:

  nslookup *File system domain name*

  If the resolved IP address is in network segment **100**, the DNS configuration is correct. If the IP address is in another network segment, the DNS configuration is incorrect. In this case, go to **b**.

b. Modify the **/etc/resolv.conf** configuration file, configure the correct tenant DNS, and run **vi /etc/resolv.conf** to edit the **/etc/resolv.conf** file. Add the DNS server IP address above the existing nameserver information.

Figure 5-1 Configuring DNS



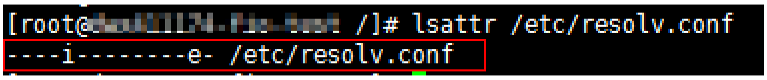
```
; generated by /sbin/dhclient-script
search openstacklocal
nameserver [redacted]
nameserver [redacted]
nameserver [redacted]
```

The format is as follows:

nameserver *DNS server IP address*

- If the configuration succeeds, go to **c**.
- If the configuration fails, run the **lsattr /etc/resolv.conf** command. If the information shown in **Figure 5-2** is displayed, the file is locked.

Figure 5-2 A locked file



```
[root@[redacted] /]# lsattr /etc/resolv.conf
----i--------e- /etc/resolv.conf
```

Run the **chattr -i/etc/resolv.conf** command to unlock the file. Then, re-configure the DNS and go to **c**.

c. Press **Esc**, input **:wq**, and press **Enter** to save the changes and exit the vi editor.

d. The default DNS of the ECS applied by the user is inherited from the VPC to which the ECS belongs. Therefore, when the ECS restarts, the ECS changes synchronously. For this reason, changing configurations of the ECS does not settle the issue completely. You need to modify configurations in the VPC. Set a correct tenant DNS for the subnet of the VPC to which the ECS belongs.

e. (Optional) Restart the server.

f. Run the **mount** command again.

- If the problem is solved, no further action is required.
- If the problem persists, see the solution for cause 4.

- Cause 4: The server where the file system is to be mounted runs Ubuntu18 or later.
    a. Reconfigure DNS by referring to **Configuring DNS**.
    b. Check whether the target server running Ubuntu18 or later uses a private image.
        - If yes, go to **d**.
        - If no, go to **c**.
    c. Convert the public image server to a private image server.
        i. To create a private image based on an existing ECS, see section "Creating an Image" in the *Elastic Cloud Server User Guide*.
        ii. Use the private image created in **c.i** to create an ECS or change the ECS OS using the private image created in **c.i**. For details, see section "Changing the OS" in the *Elastic Cloud Server User Guide*.
    d. Log in to the server and mount the file system again.

5.2 Mounting a File System Fails

Symptom

When a file system is mounted to servers using the **mount** command, message **access denied** is displayed.

Possible Causes

- Cause 1: The file system has been deleted.
- Cause 2: The server and the mounted file system are not in the same VPC.
- Cause 3: The mount point in the **mount** command is incorrect.
- Cause 4: The IP address used for accessing SFS is a virtual IP address.
- Cause 5: The DNS used for accessing the file system is incorrect.
- Cause 6: A CIFS file system is mounted to Linux servers.

Fault Diagnosis

Take troubleshooting measures based on possible causes.

Solution

- Cause 1: The file system has been deleted.

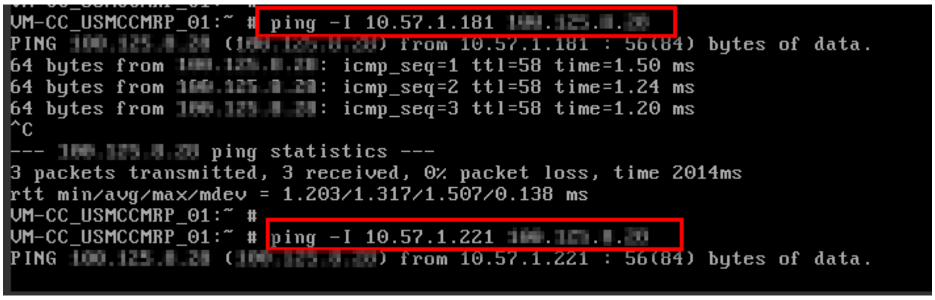
  Log in to the management console and check whether the file system has been deleted.
    - If yes, create a file system or select an existing file system to mount. Ensure that the server and the file system reside in the same VPC.
    - If no, go to Cause 2.
- Cause 2: The server and the mounted file system are not in the same VPC.

  Log in to the management console and check whether the server and the file system belong to the same VPC.

  - If yes, go to Cause 3.
  - If no, select a file system that resides in the same VPC as the server.
- Cause 3: The mount point in the **mount** command is incorrect.
  a. Log in to the management console and check whether the mount point is the same as the one in the **mount** command.
  b. If the mount point in the **mount** command is incorrectly entered, correct it and run the command again.
- Cause 4: The IP address used for accessing SFS is a virtual IP address.

  Log in to the server and run the **ping** command and use the server IP address to access SFS. Check whether the service is reachable. See **Figure 5-3**.
  - If yes, the network problem has been resolved. Check other possible causes.
  - If no, the server virtual IP address is unable to access SFS due to the network problem. Use the private IP address and run the **ping** command to access SFS and check whether the service is reachable.

Figure 5-3 Running the ping command to access SFS



```
VM-CC_USMCCMRP_01:~ # ping -I 10.57.1.181 [redacted]
PING [redacted] ([redacted]) from 10.57.1.181 : 56(84) bytes of data.
64 bytes from [redacted]: icmp_seq=1 ttl=58 time=1.50 ms
64 bytes from [redacted]: icmp_seq=2 ttl=58 time=1.24 ms
64 bytes from [redacted]: icmp_seq=3 ttl=58 time=1.20 ms
^C
--- [redacted] ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2014ms
rtt min/avg/max/mdev = 1.203/1.317/1.507/0.138 ms
VM-CC_USMCCMRP_01:~ #
VM-CC_USMCCMRP_01:~ # ping -I 10.57.1.221 [redacted]
PING [redacted] ([redacted]) from 10.57.1.221 : 56(84) bytes of data.
```

- Cause 5: The DNS used for accessing the file system is incorrect.

  Run the following command to check whether the DNS is correct:

  nslookup *File system domain name*

  Check whether the resolved IP address is in segment **100**.
  - If yes, the DNS configuration is correct. Check other possible causes.
  - If no, the DNS configuration is incorrect. Reconfigure DNS by referring to **Configuring DNS**.
- Cause 6: A CIFS file system is mounted to Linux servers.

  CIFS file systems cannot be mounted to Linux servers. Mount the CIFS file system to Windows servers.

5.3 File System Performance Is Poor

Symptom

Data is written slowly to the file system, the file system performance cannot meet service requirements, or file transfer is slow.

Troubleshooting

Possible causes are sequenced based on their occurrence probability.

If the fault persists after you have ruled out one cause, move on to the next one in the list.

Figure 5-4 Troubleshooting

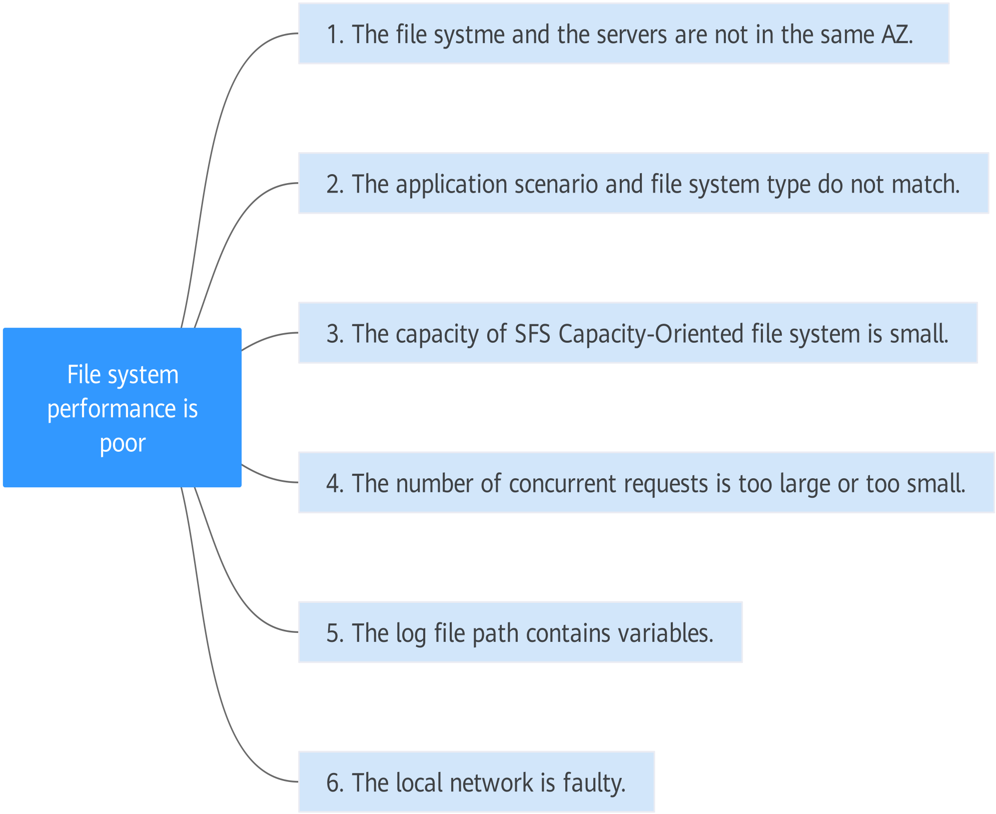

File system performance is poor
1. The file systme and the servers are not in the same AZ.
2. The application scenario and file system type do not match.
3. The capacity of SFS Capacity-Oriented file system is small.
4. The number of concurrent requests is too large or too small.
5. The log file path contains variables.
6. The local network is faulty.

Table 5-1 Troubleshooting

Possible Causes	Solution
The file system and the servers where the file system is mounted are not in the same AZ.	Create a file system in the same AZ as the servers, migrate the data from the original file system to the new file system, and mount the new file system to the servers.
The application scenario does not match the file system type.	Select an appropriate file system type based on your workloads. For details, see **File System Types**.
The capacity of the SFS Capacity-Oriented file system is small.	The performance of an SFS Capacity-Oriented file system is related to its capacity. If your workloads require a higher bandwidth, create a file system with a larger capacity.
The number of concurrent requests is too large or too small.	When the number of concurrent requests is too large or too small, the file system performance may deteriorate.

Possible Causes	Solution
The log file path contains variables.	If it takes a long time to write logs to the file system using Nginx, do as follows: ● Delete variables from the **access_log** directive and use a fixed path to store log files. ● Set the log file descriptor cache using the **open_log_file_cache** command, which improves the performance of the log path containing variables.
The local network is faulty.	Rectify the network fault.

5.4 Failed to Create an SFS Turbo File System

Symptom

An SFS Turbo file system fails to be created.

Fault Diagnosis

The following fault causes are sequenced based on their occurrence probability.

If the fault persists after you have ruled out a cause, check other causes.

Figure 5-5 Fault diagnosis

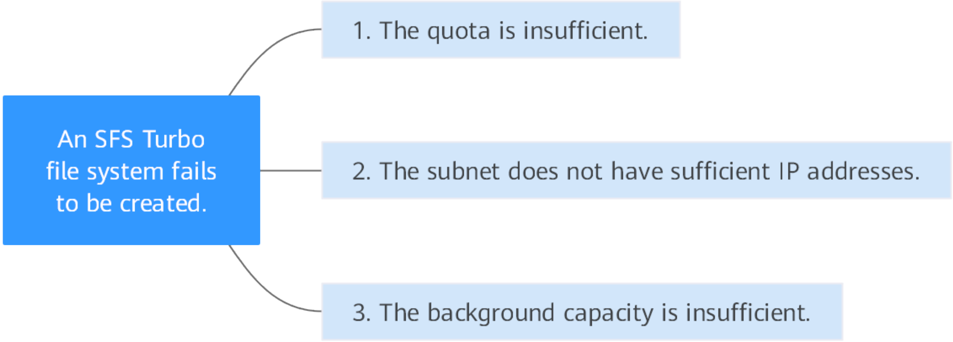

An SFS Turbo file system fails to be created.
1. The quota is insufficient.
2. The subnet does not have sufficient IP addresses.
3. The background capacity is insufficient.

Table 5-2 Fault diagnosis

Possible Cause	Solution
The quota is insufficient.	The number of created file systems has reached the upper limit. to increase the quota.
The subnet does not have sufficient IP addresses.	If the subnet IP addresses are insufficient, you can change the subnet or release other IP addresses in the subnet.

Possible Cause	Solution
The background capacity is insufficient.	to expand the capacity.

5.5 A File System Is Automatically Disconnected from the Server

Symptom

A file system is disconnected from the server and needs to be mounted again.

Possible Causes

Automatic mounting is not configured. The server is automatically disconnected from the file system after restart.

Solution

Configure automatic mounting for the server so that the file system will be automatically mounted to the server after the server restarts. For details, see .

5.6 A Server Fails to Access a File System

Symptom

A server fails to access a file system. The system displays a message indicating that the access request is denied. All services on the server are abnormal.

Possible Causes

- Cause 1: The file system is abnormal.
- Cause 2: After a forcible unmount operation on the server, mount fails.

Fault Diagnosis

Take troubleshooting measures based on possible causes.

Solution

- Cause 1: The file system is abnormal.

  Log in to the management console. On the **Scalable File System** page, check whether the file system is in the **Available** state.

  - If yes, go to Cause 2.
  - If no, see **The File System Is Abnormal** to restore the file system to the available state, and then access the file system again.

- Cause 2: After a forcible unmount operation on the server, mount fails.
  a. This problem is caused by an inherent defect of servers. Restart the server to resolve this problem.
  b. Check whether the file system can be properly mounted and accessed.
     - If yes, no further action is required.
     - If no, contact technical support.

5.7 The File System Is Abnormal

Currently, the file system exceptions include deletion error, expansion error, reduction error, and reduction failure. When the file system is in these statuses, refer to the following handling suggestions.

Table 5-3 Measures for handling file system abnormalities

Exception	Suggestion
Deletion error	When the file system is in the deletion error status, it can automatically recover to the available state. If the status cannot be restored to available, contact the administrator.
Expansion error	When the file system is in the expansion error status, it can automatically recover to the available state. If the status cannot be restored to available, contact the administrator.
Reduction error	When the file system is in the reduction error status, it takes approximately five minutes for the file system to restore to the available state.
Reduction failure	When the file system is in the reduction failure status, it takes approximately five minutes for the file system to restore to the available state.

5.8 Data Fails to Be Written into a File System Mounted to ECSs Running Different Types of Operating Systems

A file system can be mounted to a Linux ECS and a Windows ECS. However, data may fail to be written to the file system.

Symptom

If a file system is mounted to a Linux ECS and a Windows ECS, you cannot write data to the files created by the Linux ECS on the Windows ECS.

Possible Causes

A shared NFS file system belongs to the root user and cannot be modified. The write permission is granted to a user only when both the values of UID and GID of the user are **0**. You can check your UID using Windows commands. If the value of UID is, for example, **-2**, you do not have the write permission.

Fault Diagnosis

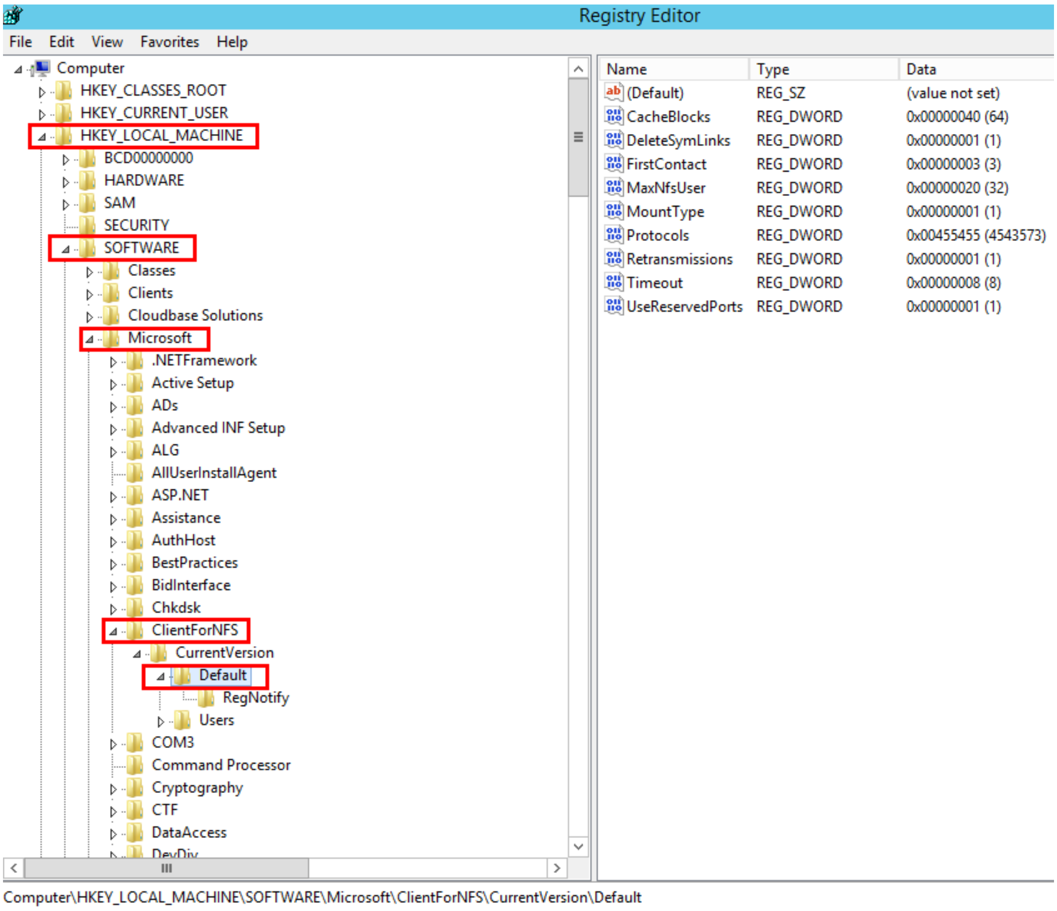
To address this problem, modify the registry and change both UID and GID values to **0** for NFS accesses from Windows.

Solution

Step 1 Choose **Start** > **Run** and enter **regedit** to open the registry.

Step 2 Enter the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default** directory. **Figure 5-6** shows an example of the directory.

Figure 5-6 Entering the directory

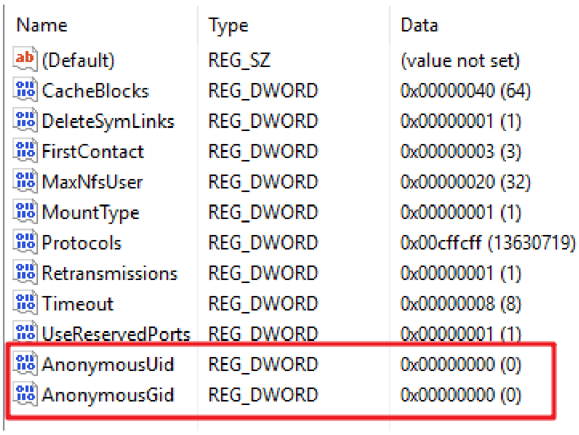


Name	Type	Data
(Default)	REG_SZ	(value not set)
CacheBlocks	REG_DWORD	0x00000040 (64)
DeleteSymLinks	REG_DWORD	0x00000001 (1)
FirstContact	REG_DWORD	0x00000003 (3)
MaxNfsUser	REG_DWORD	0x00000020 (32)
MountType	REG_DWORD	0x00000001 (1)
Protocols	REG_DWORD	0x00455455 (4543573)
Retransmissions	REG_DWORD	0x00000001 (1)
Timeout	REG_DWORD	0x00000008 (8)
UseReservedPorts	REG_DWORD	0x00000001 (1)

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ClientForNFS\CurrentVersion\Default

Step 3 Right-click the blank area and choose **New** > **DWORD Value** from the shortcut menu. Set **AnonymousUid** and **AnonymousGid** to **0**. **Figure 5-7** shows a successful operation.

Figure 5-7 Adding values



Name	Type	Data
(Default)	REG_SZ	(value not set)
CacheBlocks	REG_DWORD	0x00000040 (64)
DeleteSymLinks	REG_DWORD	0x00000001 (1)
FirstContact	REG_DWORD	0x00000003 (3)
MaxNfsUser	REG_DWORD	0x00000020 (32)
MountType	REG_DWORD	0x00000001 (1)
Protocols	REG_DWORD	0x00cffcff (13630719)
Retransmissions	REG_DWORD	0x00000001 (1)
Timeout	REG_DWORD	0x00000008 (8)
UseReservedPorts	REG_DWORD	0x00000001 (1)
AnonymousUid	REG_DWORD	0x00000000 (0)
AnonymousGid	REG_DWORD	0x00000000 (0)

Step 4 After modifying the registry, restart the server for the modification to take effect.

----End

5.9 Failed to Mount an NFS File System to a Windows IIS Server

Symptom

When an NFS file system is mounted to a Windows IIS server, an error message is displayed, indicating that the path format is not supported, and the mounting fails.

Possible Causes

The physical path of the IIS Web server is incorrect.

Fault Diagnosis

Take troubleshooting measures based on possible causes.

Solution

Step 1 Log in to the ECS. An ECS running Windows Server 2012 R2 is used in this example.

Step 2 Click **Server Manager** in the lower left corner.

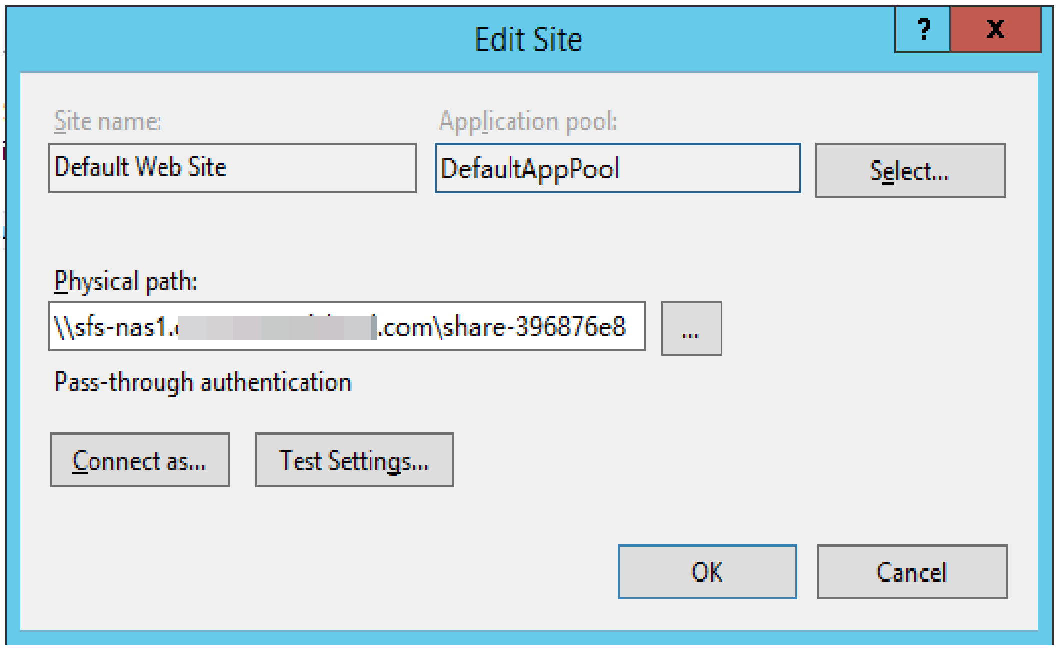
Step 3 Choose **Tools** > **Internet Information Services (IIS) Manager**, expand **Sites**, and select the target website.

Step 4 Click **Basic Settings** to check whether the **Physical path** is correct.

Step 5 The correct physical path is that of the mount point with the colon (:) deleted.

You need to enter the physical path **\\sfs-nas1.XXXXXXXXX.com\share-396876e8**, as shown in Figure 5-8.

Figure 5-8 Physical path



Site name: Default Web Site

Application pool: DefaultAppPool

Physical path: \\sfs-nas1.[redacted].com\share-396876e8

Pass-through authentication

----End

5.10 Writing to a File System Fails

Symptom

Data fails to be written to the file system mounted to ECSs running the same type of operating system.

Possible Causes

The ECS security group configuration is incorrect. The port used to communicate with the file system is not enabled.

Fault Diagnosis

Check whether the port of the target server is enabled and correctly configure the port on the security group console.

Solution

Step 1 Log in to the ECS console.

1. Log in to the management console.
2. Click 📍 in the upper left corner and select your desired region and project.
3. Under **Compute**, choose **Elastic Cloud Server**.

Step 2 In the navigation pane on the left, choose **Elastic Cloud Server**. On the page displayed, select the target server. Go to the server details page.

Step 3 Click the **Security Groups** tab and select the target security group. Click **Manage Rule** to go to the security group console.

Step 4 On the displayed page, click the **Inbound Rules** tab and click **Add Rule**. The **Add Inbound Rule** page is displayed. Add rules as follows:

After an SFS Turbo file system is created, the system automatically enables the security group port required by the NFS protocol. This ensures that the SFS Turbo file system can be accessed by your servers and prevents file system mounting failures. The inbound ports required by the NFS protocol are ports 111, 445, 2049, 2051, 2052, and 20048. If you need to change the enabled ports, choose **Access Control** > **Security Groups** of the VPC console and locate the target security group.

You are advised to use an independent security group for an SFS Turbo file system to isolate it from service nodes.

You need to add inbound and outbound rules for the security group of an SFS Capacity-Oriented file system. For details, see section "Adding a Security Group Rule" in the *Virtual Private Cloud User Guide*. In an SFS Capacity-Oriented file system, the inbound ports required by the NFS protocol are ports 111, 2049, 2051, and 2052. The inbound port required by the DNS server is port 53 and that required by the CIFS protocol is port 445.

Step 5 Click **OK**. Access the file system again for verification.

----End

5.11 Error Message "wrong fs type, bad option" Is Displayed During File System Mounting

Symptom

The message "wrong fs type, bad option" is displayed when you run the **mount** command to mount a file system to an ECS running Linux.

Possible Causes

An NFS client is not installed on the Linux ECS. That is, the **nfs-utils** software package is not installed before you execute the **mount** command.

Fault Diagnosis

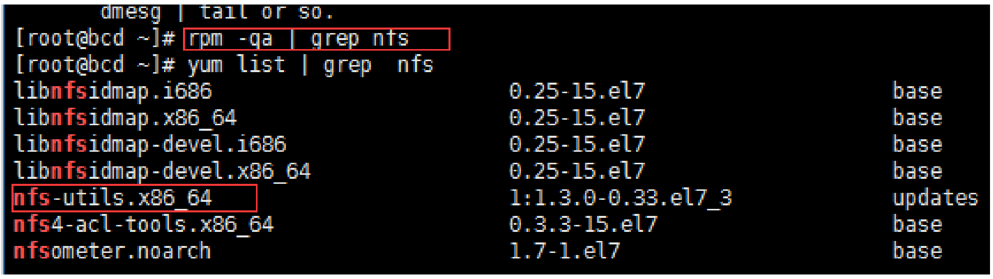
Install the required **nfs-utils** software package.

Solution

Step 1 Log in to the ECS and check whether the **nfs-utils** package is installed. Run the following command. If no command output is displayed, the package is not installed.

```
rpm -qa|grep nfs
```

Figure 5-9 Checking whether the software package has been installed

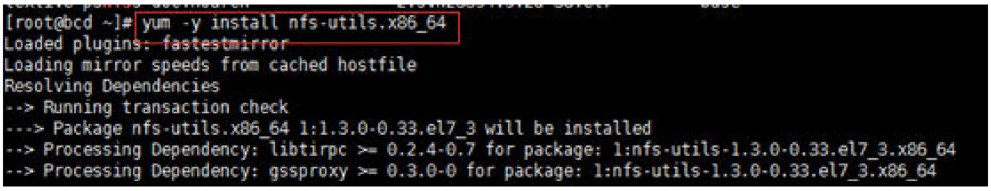


```
      dmesg | tail or so.
[root@bcd ~]# rpm -qa | grep nfs
[root@bcd ~]# yum list | grep  nfs
libnfsidmap.i686                      0.25-15.el7                  base
libnfsidmap.x86_64                    0.25-15.el7                  base
libnfsidmap-devel.i686                0.25-15.el7                  base
libnfsidmap-devel.x86_64              0.25-15.el7                  base
nfs-utils.x86_64                      1:1.3.0-0.33.el7_3           updates
nfs4-acl-tools.x86_64                 0.3.3-15.el7                 base
nfsometer.noarch                      1.7-1.el7                    base
```

Step 2 Run the following command to install the nfs-utils software package:

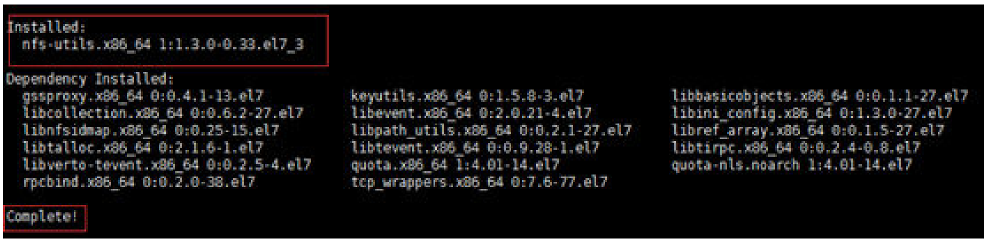
```
yum -y install nfs-utils
```

Figure 5-10 Executing the installation command



```
[root@bcd ~]# yum -y install nfs-utils.x86_64
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package nfs-utils.x86_64 1:1.3.0-0.33.el7_3 will be installed
--> Processing Dependency: libtirpc >= 0.2.4-0.7 for package: 1:nfs-utils-1.3.0-0.33.el7_3.x86_64
--> Processing Dependency: gssproxy >= 0.3.0-0 for package: 1:nfs-utils-1.3.0-0.33.el7_3.x86_64
```

Figure 5-11 Successful installation



```
Installed:
  nfs-utils.x86_64 1:1.3.0-0.33.el7_3

Dependency Installed:
  gssproxy.x86_64 0:0.4.1-13.el7            keyutils.x86_64 0:1.5.8-3.el7               libbasicobjects.x86_64 0:0.1.1-27.el7
  libcollection.x86_64 0:0.6.2-27.el7       libevent.x86_64 0:2.0.21-4.el7              libini_config.x86_64 0:1.3.0-27.el7
  libnfsidmap.x86_64 0:0.25-15.el7          libpath_utils.x86_64 0:0.2.1-27.el7         libref_array.x86_64 0:0.1.5-27.el7
  libtalloc.x86_64 0:2.1.6-1.el7            libtevent.x86_64 0:0.9.28-1.el7             libtirpc.x86_64 0:0.2.4-0.8.el7
  libverto-tevent.x86_64 0:0.2.5-4.el7      quota.x86_64 1:4.01-14.el7                  quota-nls.noarch 1:4.01-14.el7
  rpcbind.x86_64 0:0.2.0-38.el7             tcp_wrappers.x86_64 0:7.6-77.el7

Complete!
```

Step 3 Run the **mount** command again to mount the file system to the ECS.

mount -t nfs -o vers=3,timeo=600,noresvport,nolock *Mount point Local path*

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

mount -l

If the command output contains the following information, the file system is mounted successfully.

```
example.com:/share-xxx on /local_path type nfs (rw,vers=3,timeo=600,nolock,addr=)
```

----End

5.12 Failed to Access the Shared Folder in Windows

Symptom

When you mount a file system to an ECS running Windows, the system displays a message "You cannot access this shared folder because your organization's security policies block unauthenticated guest access. These policies help to protect you PC from unsafe or malicious devices on the network."

Possible Causes

Guest access to CIFS file systems is blocked or disabled.

Fault Diagnosis

Solution 1: Manually enable guest access.

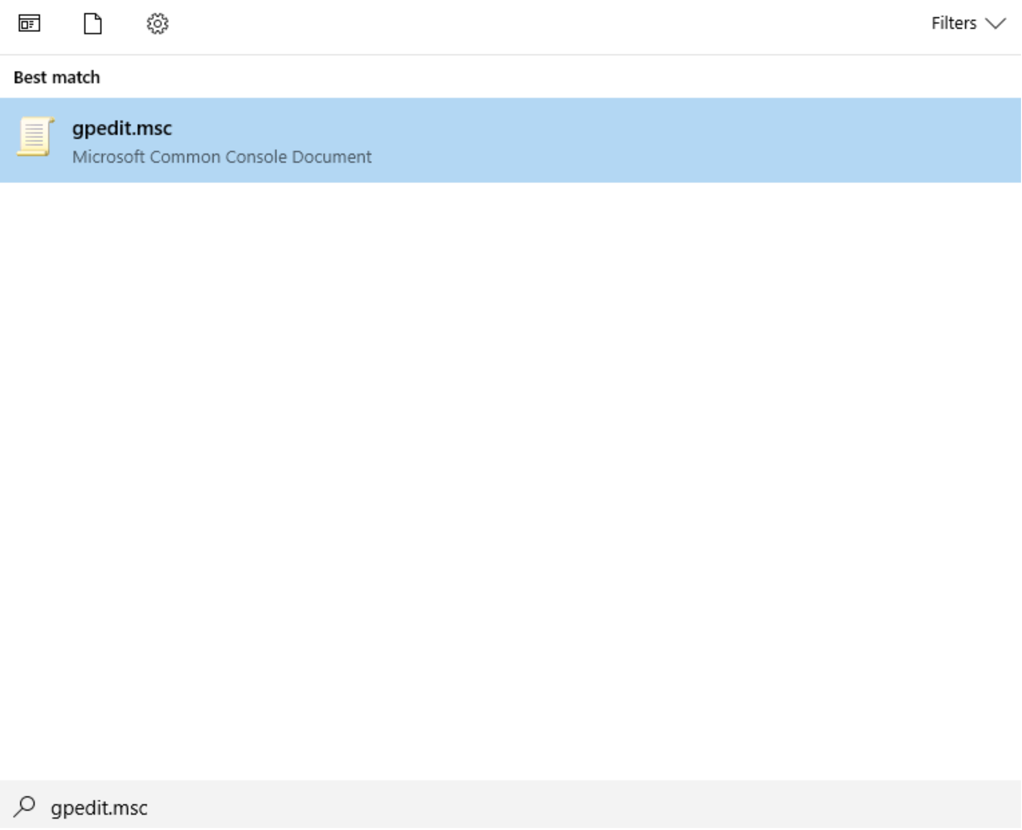
Solution 2: Modify the registry to allow guest access (suitable for versions later than Windows Server 2016).

Solution

Solution 1: Manually enable guest access.

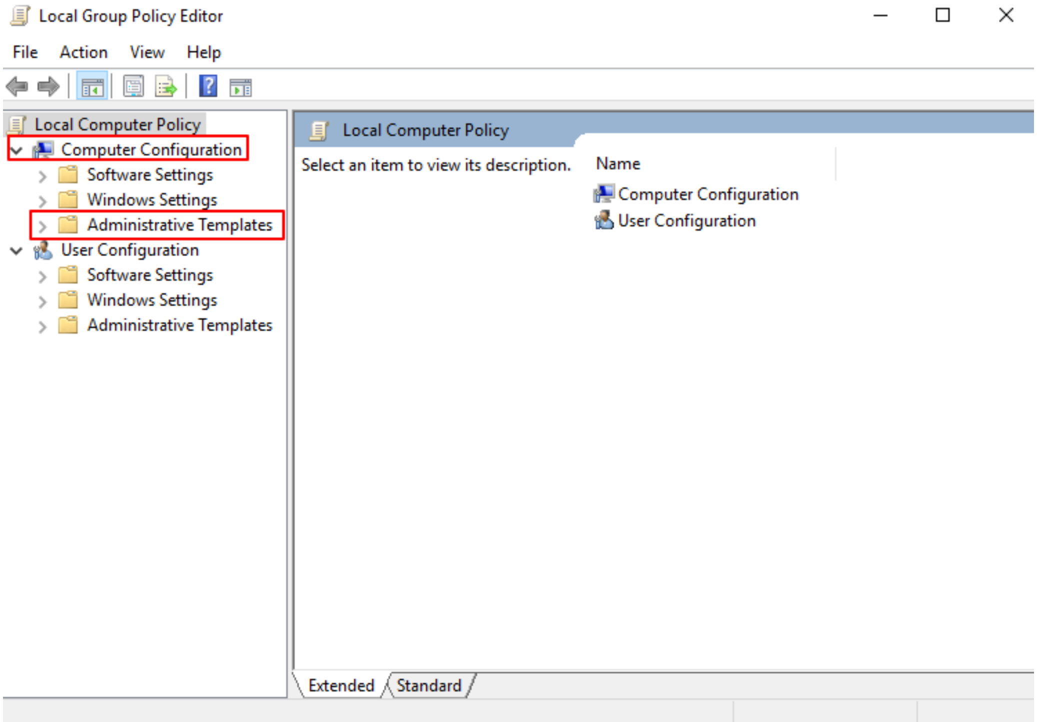
Step 1 Open **Run** command box, enter **gpedit.msc**, and press **Enter** to start **Local Group Policy Editor**.

Figure 5-12 Entering gpedit.msc



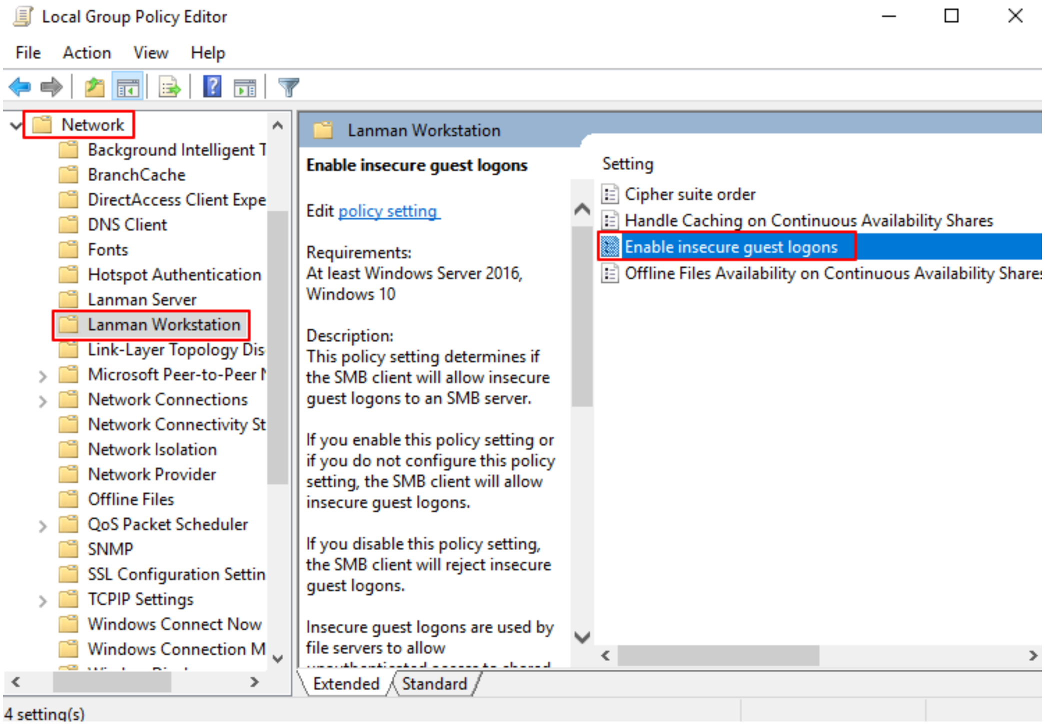
Step 2 On the **Local Group Policy Editor** page, choose **Computer Configuration** > **Administrative Templates**.

Figure 5-13 Local Group Policy Editor



Step 3 Under **Administrative Templates**, choose **Network** > **Lanman Workstation** and find the option of **Enable insecure guest logons**.

Figure 5-14 Locating the option



Enable insecure guest logons

Edit policy setting

Requirements:
At least Windows Server 2016, Windows 10

Description:
This policy setting determines if the SMB client will allow insecure guest logons to an SMB server.

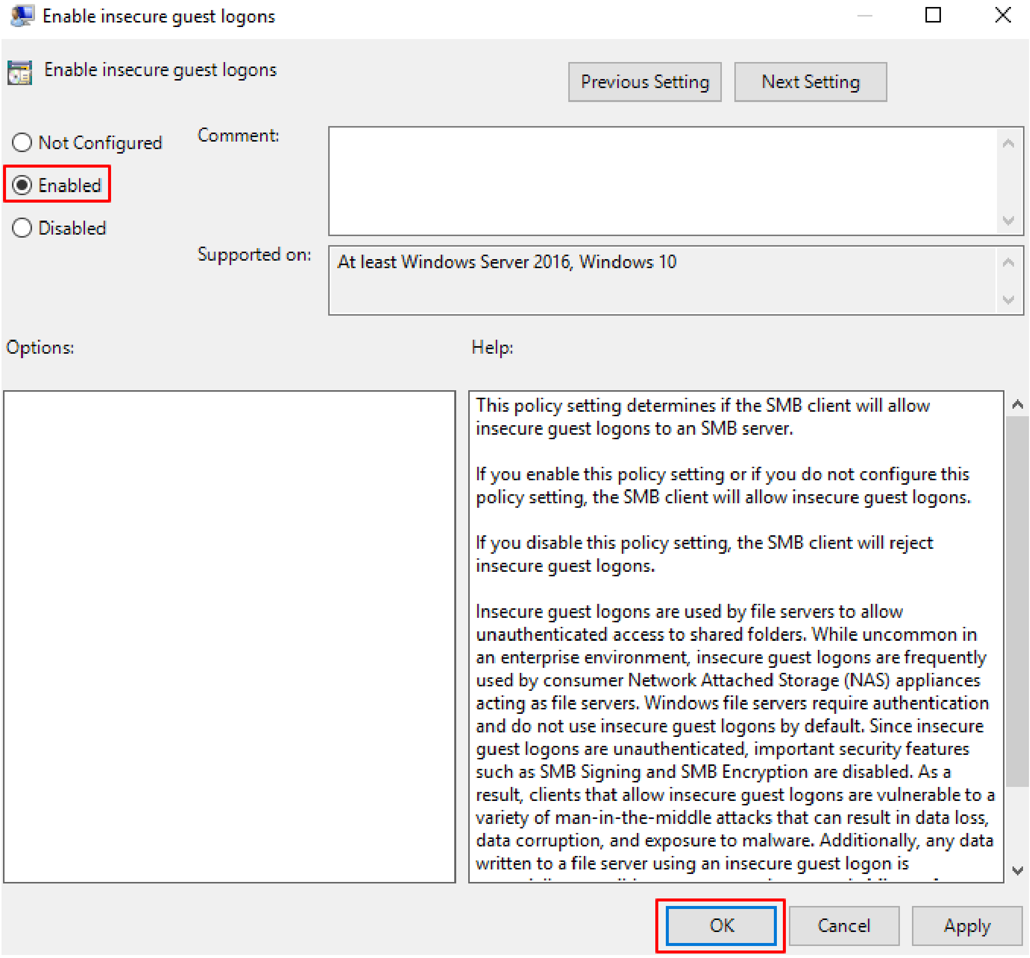
If you enable this policy setting or if you do not configure this policy setting, the SMB client will allow insecure guest logons.

If you disable this policy setting, the SMB client will reject insecure guest logons.

Insecure guest logons are used by file servers to allow

Step 4 Double-click **Enable insecure guest logons**. Select **Enabled** and click **OK**.

Figure 5-15 Enabling insecure guest logons



Enable insecure guest logons

Enable insecure guest logons

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Windows Server 2016, Windows 10

Options:

Help:

This policy setting determines if the SMB client will allow insecure guest logons to an SMB server.

If you enable this policy setting or if you do not configure this policy setting, the SMB client will allow insecure guest logons.

If you disable this policy setting, the SMB client will reject insecure guest logons.

Insecure guest logons are used by file servers to allow unauthenticated access to shared folders. While uncommon in an enterprise environment, insecure guest logons are frequently used by consumer Network Attached Storage (NAS) appliances acting as file servers. Windows file servers require authentication and do not use insecure guest logons by default. Since insecure guest logons are unauthenticated, important security features such as SMB Signing and SMB Encryption are disabled. As a result, clients that allow insecure guest logons are vulnerable to a variety of man-in-the-middle attacks that can result in data loss, data corruption, and exposure to malware. Additionally, any data written to a file server using an insecure guest logon is

Step 5 After the access is enabled, mount the file system again. If the fault persists, contact technical support.

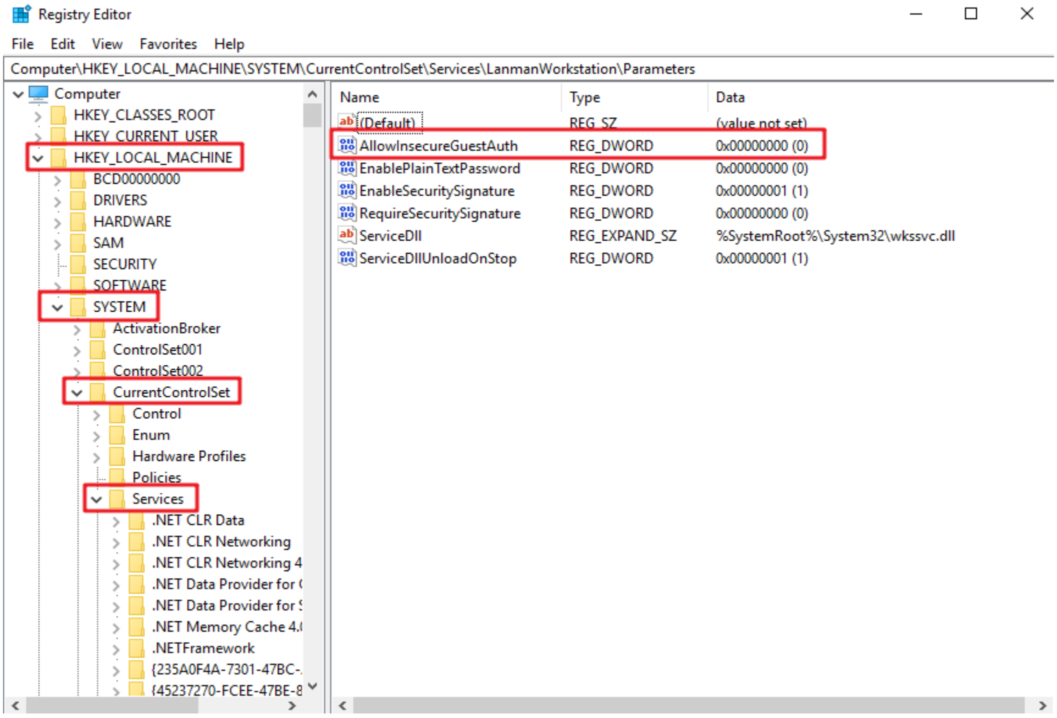
----End

Solution 2: Modify the registry to allow guest access (suitable for versions later than Windows Server 2016).

Step 1 Choose **Start** > **Run** and enter **regedit** to open the registry.

Step 2 Go to the **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Parameters** directory.

Figure 5-16 Entering the registry

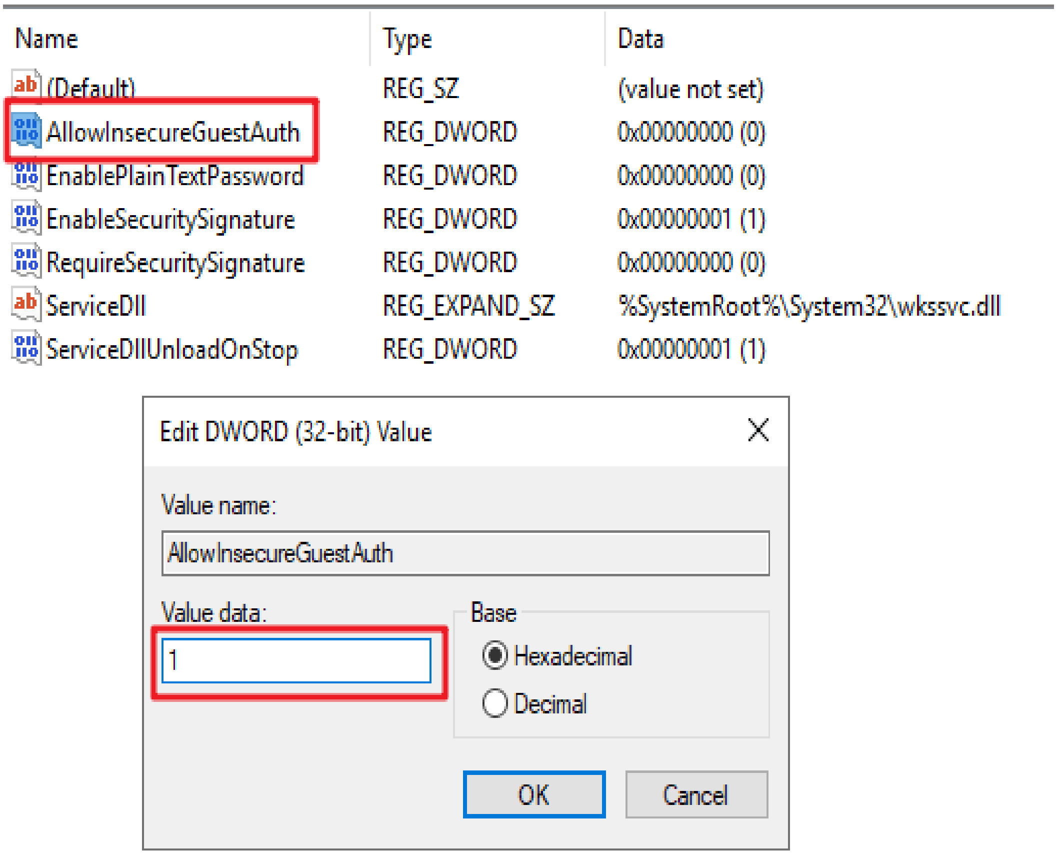


Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Parameters

Name	Type	Data
(Default)	REG_SZ	(value not set)
AllowInsecureGuestAuth	REG_DWORD	0x00000000 (0)
EnablePlainTextPassword	REG_DWORD	0x00000000 (0)
EnableSecuritySignature	REG_DWORD	0x00000001 (1)
RequireSecuritySignature	REG_DWORD	0x00000000 (0)
ServiceDll	REG_EXPAND_SZ	%SystemRoot%\System32\wkssvc.dll
ServiceDllUnloadOnStop	REG_DWORD	0x00000001 (1)

Step 3 Right-click **AllowInsecureGuestAuth** and choose **Modify** from the shortcut menu. In the pop-up window, change the value to **1**.

Figure 5-17 Changing the value



Name	Type	Data
(Default)	REG_SZ	(value not set)
AllowInsecureGuestAuth	REG_DWORD	0x00000000 (0)
EnablePlainTextPassword	REG_DWORD	0x00000000 (0)
EnableSecuritySignature	REG_DWORD	0x00000001 (1)
RequireSecuritySignature	REG_DWORD	0x00000000 (0)
ServiceDll	REG_EXPAND_SZ	%SystemRoot%\System32\wkssvc.dll
ServiceDllUnloadOnStop	REG_DWORD	0x00000001 (1)

Edit DWORD (32-bit) Value

Value name: AllowInsecureGuestAuth

Value data: 1

Base

● Hexadecimal

○ Decimal

----End

6 FAQs

6.1 Concepts

6.1.1 What Is SFS?

Scalable File Service (SFS) provides scalable, high-performance file storage. With SFS, you can enjoy shared file access spanning multiple ECSs. SFS supports the Network File System (NFS) protocol. You can seamlessly integrate existing applications and tools with the service.

SFS provides an easy-to-use graphical user interface (GUI). On the GUI, users can create and configure file systems, saving effort in deploying, resizing, and optimizing file systems.

In addition, SFS features high reliability and availability. It can be elastically expanded, and it performs better as its capacity grows. The service is suitable for a wide range of scenarios, including media processing, file sharing, content management and web services, big data, and analytic applications.

6.1.2 What Is SFS Turbo?

SFS Turbo provides high-performance file storage that can be expanded on demand. With SFS Turbo, you can enjoy shared file access spanning multiple ECSs. SFS Turbo supports the Network File System (NFS) protocol (only NFSv3). You can seamlessly integrate existing applications and tools with the service.

SFS Turbo provides an easy-to-use graphical user interface (GUI). On the GUI, users can create and configure file systems, saving effort in deploying, resizing, and optimizing file systems.

In addition, SFS Turbo features high reliability and availability. It can be elastically expanded, and it performs better as its capacity grows. The service is suitable for a wide range of scenarios, including enterprise office, high-performance websites, and software development. For details about the file system types, see **File System Types**.

6.1.3 What Are the Differences Between SFS, OBS, and EVS?

Table 6-1 shows the comparison between SFS, OBS, and EVS.

Table 6-1 Comparison between SFS, OBS, and EVS

Dimension	SFS	OBS	EVS
Concept	SFS provides on-demand high-performance file storage, which can be shared by multiple ECSs. 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 various 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.
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 Scenario	Gene sequencing, image rendering, media processing, file sharing, content management, and web services	Big data analysis, 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	PB-scale	EB-scale	TB-scale
Latency	3–10 ms	10 ms	1 ms

Dimension	SFS	OBS	EVS
IOPS/TPS	10,000 for a single file system	Tens of millions	50,000 for a single disk
Bandwidth	GB/s	TB/s	MB/s
Data sharing	Supported	Supported	Supported
Remote access	Supported	Supported	Not supported
Online editing	Supported	Not supported	Supported
Used independently	Supported	Supported	Not supported

6.2 Specifications

6.2.1 What Is the Maximum Size of a File That Can Be Stored in a File System?

For SFS Capacity-Oriented file systems, the maximum supported size of a file is 240 TB.

For SFS Turbo file systems, the maximum supported size of a file is 16 TB.

6.2.2 What Access Protocols Are Supported by SFS?

SFS Capacity-Oriented supports standard NFSv3 and CIFS. SFS Turbo supports the standard NFSv3 protocol.

6.2.3 How Many File Systems Can Be Created by Each Account?

Each account can create a maximum of 10 SFS Capacity-Oriented file systems and 10 SFS Turbo file systems.

- SFS Capacity-Oriented file systems can be created in batches. To create more than 10 SFS Capacity-Oriented file systems, click **Increase quota** on the page for creating a file system.
- Only one SFS Turbo file system can be created at a time. To create more than 10 SFS Turbo file systems, contact customer service to apply for a higher quota.

6.2.4 How Many Servers Can a File System Be Mounted To?

You can mount an SFS Capacity-Oriented file system to a maximum of 10,000 servers.

You can mount an SFS Turbo file system to a maximum of 3,000 servers.

6.3 Restrictions

6.3.1 Can the Capacity of a File System Be Expanded?

Yes, by capacity resizing.

6.3.2 Can I Migrate My File System Data to Another Region?

Cross-region migration of file system data is currently not supported. It is recommended that you select an appropriate region when purchasing a file system. Alternatively, you can copy the data to a local device and transfer it to another region.

If you are using SFS Turbo file systems, you can back up your file system data and replicate the backups to another region using the CBR service. Then, create new SFS Turbo file systems from the backups. This way, your file system data has been migrated to another region.

6.3.3 Can a File System Be Mounted to Multiple Accounts?

Multi-account mounting for SFS Capacity-Oriented file systems is available. For details, see **Configuring Multi-Account Access**.

With VPC peering, an SFS Turbo file system can be accessed across accounts. For details about VPC peering connection and usage instructions, see section "VPC Peering Connection" in *Virtual Private Cloud User Guide*.

6.4 Networks

6.4.1 Can a File System Be Accessed Across VPCs?

Yes.

- Multi-VPC access can be configured for an SFS Capacity-Oriented file system so that ECSs in different VPCs can share the same file system, as long as the VPCs that the ECSs belong to are added to the VPC list of the file system or the ECS IP addresses are added as authorized IP addresses of the VPCs. For details, see **Configuring Multi-VPC Access**.
- An SFS Turbo file system allows two or more VPCs in the same region to interconnect with each other through VPC peering connection. In this case, different VPCs are in the same network, and ECSs in these VPCs can share the same file system. For details about VPC peering connection, see section "VPC Peering Connection" in *Virtual Private Cloud User Guide*.

6.4.2 Does SFS Support Cross-Region Mounting?

Currently, SFS does not support cross-region mounting. A file system can be mounted only to ECSs in the same region.

6.4.3 Does the Security Group of a VPC Affect SFS?

A security group is a collection of access control rules for servers that have the same security protection requirements and are mutually trusted in a VPC. After a security group is created, you can create different access rules for the security group to protect the servers that are added to this security group. The default security group rule allows all outgoing data packets. Servers in a security group can access each other without the need to add rules. The system creates a security group for each cloud account by default. Users can also create custom security groups by themselves.

After an SFS Turbo file system is created, the system automatically enables the security group port required by the NFS protocol. This ensures that the SFS Turbo file system can be accessed by your servers and prevents file system mounting failures. The inbound ports required by the NFS protocol are ports 111, 445, 2049, 2051, 2052, and 20048. If you need to change the enabled ports, choose **Access Control** > **Security Groups** of the VPC console and locate the target security group.

You are advised to use an independent security group for an SFS Turbo file system to isolate it from service nodes.

You need to add inbound and outbound rules for the security group of an SFS Capacity-Oriented file system. For details, see section "Adding a Security Group Rule" in the *Virtual Private Cloud User Guide*. In an SFS Capacity-Oriented file system, the inbound ports required by the NFS protocol are ports 111, 2049, 2051, and 2052. The inbound port required by the DNS server is port 53 and that required by the CIFS protocol is port 445.

Example Value

- Inbound rule

Direction	Protocol	Port Range	Source IP Address		Description
Inbound	TCP and UDP	111	IP Address	0.0.0.0/0 (configurable)	One port corresponds to one access rule. You need to add information to the ports one by one.

- Outbound rule

Direction	Protocol	Port Range	Source IP Address		Description
Outbound	TCP and UDP	111	IP Address	0.0.0.0/0 (configurable)	One port corresponds to one access rule. You need to add information to the ports one by one.

NOTE

The bidirectional access rule must be configured for port 111. The inbound rule can be set to the front-end service IP range of SFS. You can obtain it by running the following command: **ping** *File system domain name or IP address* or **dig** *File system domain name or IP address*.

For ports 445, 2049, 2050, 2051, and 2052, only the outbound rule needs to be added, which is the same as the outbound rule of port 111.

For the NFS protocol, add an inbound rule to open the TCP&UDP port 111, TCP ports 2049, 2051, and 2052, and UDP&TCP port 20048. For the SMB protocol, add an inbound rule to open TCP port 445.

For the NFS protocol with UDP port 20048 not opened, the time required for mounting may become longer. In this case, you can use the **-o tcp** option in **mount** to avoid this issue.

6.4.4 What Can I Do If the Data of the File System That Is Mounted to Two Servers Is Not Synchronized?

Symptom

When file system C is mounted to both server A and server B, there is a delay in synchronizing the file to server B after it is uploaded to server A. However, there is no delay when the file is uploaded to server B separately.

Fault Diagnosis

Add **noac, lookupcache=none** to the mount command.

The **noac** option disables file attribute caching and forces write synchronization. By default, an NFS client's file attribute information is cached using the **ac** option to improve performance, and the client checks file attribute information periodically and updates it if there are any changes. Within the cache validity period, the client does not check whether file attribute information on the server is changed. By default, the value of this option is **ac**. Set it to **noac**.

The **lookupcache** option is related to directory entry caching, and the value can be **all**, **none**, **pos**, or **positive**. With **lookupcache=none**, the client neither trust the positive nor negative lookup results. In this way, lookup caching is disabled.

Solution

Step 1 Unmount the file system if it has been mounted. For details, see**Unmount a File System**.

Step 2 Prepare for the mount by referring to **Mounting an NFS File System to ECSs (Linux)**.

Step 3 Run the following command to mount the file system:

```
mount -t nfs -o vers=3,timeo=600,noac,lookupcache=none,noresvport,nolock Shared path Local path
```

----End

6.5 Others

6.5.1 How Do I Access a File System from a Server?

To access your file system, install the NFS client on a Linux server and run the **mount** command to mount the file system. For a Windows server, install the NFS client, modify the NFS transfer protocol, and run the **mount** command to mount the file system. Alternatively, directly enter the mount point of the CIFS file system as an authorized user to mount the CIFS file system. Then, you can share the files and directories of the file system.

6.5.2 How Do I Check Whether a File System on a Linux Server Is Available?

Log in to the server as the **root** user. Run the following command to list all available file systems with the specified domain name or IP address:

showmount -e *File system domain name or IP address*

6.5.3 What Resources Does SFS Occupy?

To ensure that file systems can be used properly, the service occupies the following resources:

- For SFS Capacity-Oriented file systems:
  - When a file system is created, the inbound rules of ports 111, 445, 2049, 2051, and 2052 are enabled in the security group entered by the user. The default source IP address is 0.0.0.0/0. You can change the IP address as required.
  - If an encrypted SFS Capacity-Oriented file system is created, the KMS key entered by the user is used for encryption. Note that if the key is deleted, data in the file system cannot be used.
- For SFS Turbo file systems:
  - When an SFS Turbo file system is created, two private IP addresses and one virtual IP address are created in the subnet entered by the user.
  - When an SFS Turbo file system is created, the inbound rules of ports 111, 445, 2049, 2051, 2052, and 20048 are enabled in the security group entered by the user. The default source IP address is 0.0.0.0/0. You can change the IP address as required.

When data is written to the folders of a file system, the running memory of the server is occupied, but the storage space of the server disk is not occupied. The file system uses independent space.

6.5.4 Why Is the Capacity Displayed as 10P After I Mount My SFS Capacity-Oriented File System?

The size of an existing SFS Capacity-Oriented file system with automatic capacity expansion enabled is unlimited. When you run the **df -h** command on the client, the system returns **10P** for display purposes.

6.5.5 Can a File System Be Accessed Across Multiple AZs?

1. A single file system can be created only in one AZ, for example, **AZ 1**, but can be mounted to and accessed from any AZ.
2. A file system does not support data redundancy across AZs. If the AZ where a file system resides is unavailable, the file system is unavailable.

6.5.6 Can I Upgrade an SFS Capacity-Oriented File System to an SFS Turbo File System?

No. If data in the SFS Capacity-Oriented file system is no longer required, delete or unsubscribe from it and then purchase an SFS Turbo file system. If you still need the data in the SFS Capacity-Oriented file system, purchase an SFS Turbo file system and **migrate data to the SFS Turbo file system**. After the data is migrated, delete or unsubscribe from the SFS Capacity-Oriented file system.

6.5.7 Can I Upgrade an SFS Turbo File System from Standard to Standard-Enhanced?

No. The type of an existing SFS Turbo file system cannot be changed.

If you want an SFS Turbo file system of another type, delete or unsubscribe from the current file system and purchase a new one with your desired type. If you still need the data in your old file system, purchase a new file system and **migrate data to the new file system**. After the data is migrated, delete or unsubscribe from the old file system.

6.5.8 How Can I Migrate Data Between SFS and EVS?

Mount a file system and an EVS disk to the same ECS, and then manually replicate data between the file system and EVS disk.

6.5.9 Can I Directly Access SFS from On-premises Devices?

SFS Turbo supports on-premises access via Direct Connect or other methods. After network communication is established, you can access an SFS Turbo file system from your on-premises devices.

6.5.10 How Do I Delete .nfs Files?

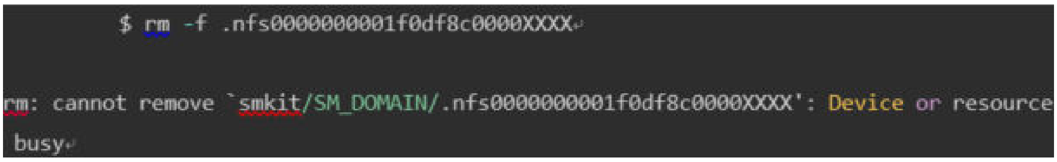
NFS .nfs Files

The .nfs files are temporary files in NFS. If you try to delete a file, and the file is still open, a .nfs file will appear. The .nfs files are used by NFS clients to manage the deletion of open files in the file system. If one process deletes a file while another process still has it open, the client will rename the file to .nfsxxx. If the last open to this file is closed, the client will automatically delete the file. If the client crashes before the file is cleared, the file will be left in the file system.

Clearing .nfs Files

The .nfs files need to be cleared. You can run the **rm -f** command to delete them. The file system will not be affected by the deletion. If an error is reported when you delete a .nfs file, do as follows:

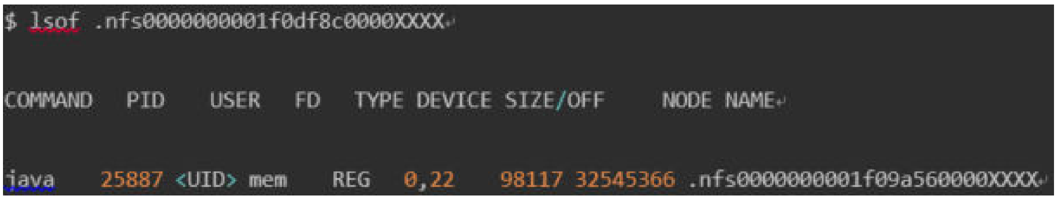
Figure 6-1 Deletion error



```
        $ rm -f .nfs0000000001f0df8c0000XXXX

rm: cannot remove `smkit/SM_DOMAIN/.nfs0000000001f0df8c0000XXXX': Device or resource
 busy
```

Run the **lsof** command to obtain the ID of the process that has the file open.

Figure 6-2 Viewing the process ID



```
$ lsof .nfs0000000001f0df8c0000XXXX

COMMAND   PID   USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME

java    25887 <UID> mem    REG   0,22    98117 32545366 .nfs0000000001f09a560000XXXX
```

If the process can be stopped, run the **kill -9** *Process ID* command to stop the process and then delete the file.

6.5.11 Why My File System Used Space Increases After I Migrate from SFS Capacity-Oriented to SFS Turbo?

SFS Turbo file systems contain metadata, which occupies about 8% to 10% file system space. That is why the used space of your file system increases after a data migration from SFS Capacity-Oriented file systems to SFS Turbo file systems. The metadata consists of the file system management data, such as the file size, file system owner, and file modification time.

6.5.12 How Can I Improve the Copy and Delete Efficiency with an SFS Turbo File System?

Common Linux commands, such as **cp**, **rm**, and **tar**, are executed sequentially. To take the concurrency advantage of cloud file systems, run commands concurrently to improve efficiency.

6.5.13 How Do Second- and Third-level Directory Permissions of an SFS Turbo File System Be Inherited?

Subdirectories in SFS Turbo file systems cannot inherit permissions of their parent directories.

6.5.14 How Do I Deploy SFS Turbo on CCE?

Complete the deployment on the CCE console based on your services by referring to section "Storage" or "Storage (FlexVolume)" in the *Cloud Container Engine User Guide*.

7 Other Operations

7.1 Testing SFS Turbo Performance

fio is an open-source I/O pressure testing tool. You can use fio to test the throughput and IOPS of SFS.

Prerequisites

fio has been installed on the ECS. It can be downloaded from **the official website** or from **GitHub**.

Note and Description

The test performance depends on the network bandwidth between the client and server, as well as the capacity of the file system.

Installing fio

The following uses a Linux CentOS system as an example:

1. Download fio.

   yum install fio

2. Install the libaio engine.

   yum install libaio-devel

3. Check the fio version.

   fio --version

File System Performance Data

The performance metrics of SFS Turbo file systems include IOPS and throughput. For details, see **Table 7-1**.

Table 7-1 File system performance data

Parameter	General	
	SFS Turbo Standard	SFS Turbo Performance
Maximum capacity	32 TB	32 TB
Maximum IOPS	5,000	20,000
Maximum throughput	150 MB/s	350 MB/s
Formula used to calculate the IOPS	IOPS = Min. [5,000, (1,200 + 6 x Capacity)] Unit: GB	IOPS = Min. [20,000, (1,500 + 20 x Capacity)] Unit: GB

IOPS Calculation Formula

- IOPS of a single file system = Min. [Maximum IOPS, (Baseline IOPS + IOPS per GB x Capacity)]

  For an SFS Turbo Performance file system:

  - If the file system capacity is 500 GB: IOPS = Min. [20,000, (1,500 + 20 x 500)] = 11,500
  - If the file system capacity is 1,000 GB: IOPS = Min. [20,000, (1,500 + 20 x 1,000)] = 20,000
- No performance calculation formula is available for the SFS Turbo Standard - Enhanced and SFS Turbo Performance - Enhanced file systems. The IOPS of an SFS Turbo Standard - Enhanced file system is 15,000, and that of an SFS Turbo Performance - Enhanced file system is 100,000.

Common Test Configuration Example

NOTE

The following estimated values are obtained from the test on a single ECS. You are advised to use multiple ECSs to test the performance of SFS.

In the following examples, SFS Turbo Performance and ECSs with the following specifications are used for illustration.

Specifications: General computing-plus | c3.xlarge.4 | 4 vCPUs | 16 GB

Image: CentOS 7.5 64-bit

Mixed read/write with a read/write ratio of 7:3

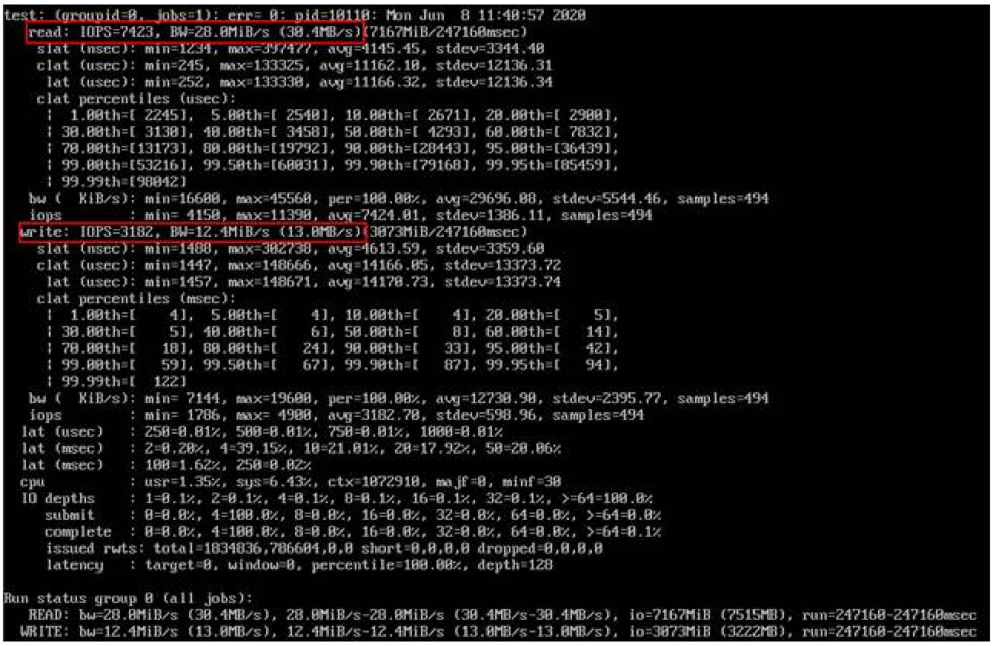
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/nfs/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=rw --rwmixwrite=30 --fallocate=none

NOTE

/mnt/nfs/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/nfs** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=10110: Mon Jun  8 11:40:57 2020
   read: IOPS=7423, BW=28.0MiB/s (30.4MB/s)(7167MiB/247160msec)
    slat (nsec): min=1234, max=397477, avg=4145.45, stdev=3344.40
    clat (usec): min=245, max=133325, avg=11162.10, stdev=12136.31
     lat (usec): min=252, max=133330, avg=11166.32, stdev=12136.34
    clat percentiles (usec):
     |  1.00th=[ 2245],  5.00th=[ 2540], 10.00th=[ 2671], 20.00th=[ 2900],
     | 30.00th=[ 3130], 40.00th=[ 3458], 50.00th=[ 4293], 60.00th=[ 7832],
     | 70.00th=[13173], 80.00th=[19792], 90.00th=[20443], 95.00th=[36439],
     | 99.00th=[53216], 99.50th=[60031], 99.90th=[79168], 99.95th=[85459],
     | 99.99th=[98042]
   bw (  KiB/s): min=16600, max=45560, per=100.00%, avg=29696.08, stdev=5544.46, samples=494
   iops        : min= 4150, max=11390, avg=7424.01, stdev=1386.11, samples=494
  write: IOPS=3182, BW=12.4MiB/s (13.0MB/s)(3073MiB/247160msec)
    slat (nsec): min=1400, max=302730, avg=4613.59, stdev=3359.60
    clat (usec): min=1447, max=148666, avg=14166.05, stdev=13373.72
     lat (usec): min=1457, max=148671, avg=14170.73, stdev=13373.74
    clat percentiles (msec):
     |  1.00th=[    4],  5.00th=[    4], 10.00th=[    4], 20.00th=[    5],
     | 30.00th=[    5], 40.00th=[    6], 50.00th=[    8], 60.00th=[   14],
     | 70.00th=[   18], 80.00th=[   24], 90.00th=[   33], 95.00th=[   42],
     | 99.00th=[   59], 99.50th=[   67], 99.90th=[   87], 99.95th=[   94],
     | 99.99th=[  122]
   bw (  KiB/s): min= 7144, max=19600, per=100.00%, avg=12730.90, stdev=2395.77, samples=494
   iops        : min= 1786, max= 4900, avg=3182.70, stdev=598.96, samples=494
  lat (usec)   : 250=0.01%, 500=0.01%, 750=0.01%, 1000=0.01%
  lat (msec)   : 2=0.20%, 4=39.15%, 10=21.01%, 20=17.92%, 50=20.06%
  lat (msec)   : 100=1.62%, 250=0.02%
  cpu          : usr=1.35%, sys=6.43%, ctx=1072910, majf=0, minf=30
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued rwts: total=1834836,786604,0,0 short=0,0,0,0 dropped=0,0,0,0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: bw=28.0MiB/s (30.4MB/s), 28.0MiB/s-28.0MiB/s (30.4MB/s-30.4MB/s), io=7167MiB (7515MB), run=247160-247160msec
  WRITE: bw=12.4MiB/s (13.0MB/s), 12.4MiB/s-12.4MiB/s (13.0MB/s-13.0MB/s), io=3073MiB (3222MB), run=247160-247160msec
```

Mixed read/write with a read/write ratio of 3:7

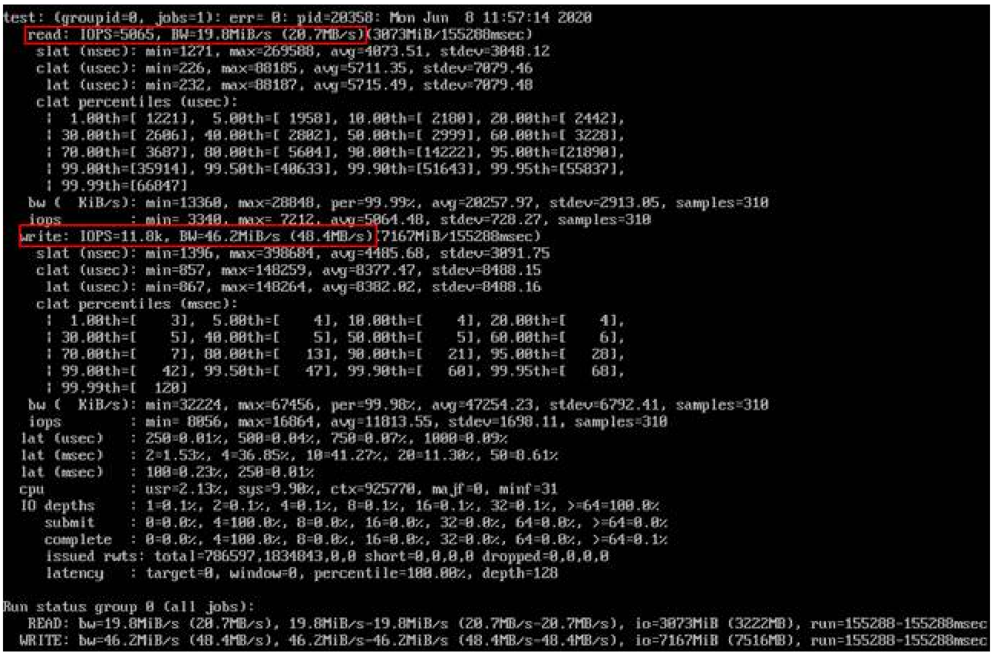
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/nfs/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=rw --rwmixwrite=70 --fallocate=none

NOTE

/mnt/nfs/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/nfs** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=20358: Mon Jun  8 11:57:14 2020
  read: IOPS=5065, BW=19.8MiB/s (20.7MB/s)(3073MiB/155288msec)
   slat (nsec): min=1271, max=269588, avg=4073.51, stdev=3048.12
   clat (usec): min=226, max=88185, avg=5711.35, stdev=7079.46
    lat (usec): min=232, max=88187, avg=5715.49, stdev=7079.48
   clat percentiles (usec):
    |  1.00th=[ 1221],  5.00th=[ 1958], 10.00th=[ 2180], 20.00th=[ 2442],
    | 30.00th=[ 2606], 40.00th=[ 2802], 50.00th=[ 2999], 60.00th=[ 3228],
    | 70.00th=[ 3687], 80.00th=[ 5604], 90.00th=[14222], 95.00th=[21890],
    | 99.00th=[35914], 99.50th=[40633], 99.90th=[51643], 99.95th=[55837],
    | 99.99th=[66847]
  bw (  KiB/s): min=13360, max=28848, per=99.99%, avg=20257.97, stdev=2913.05, samples=310
  iops        : min= 3340, max= 7212, avg=5064.48, stdev=728.27, samples=310
 write: IOPS=11.8k, BW=46.2MiB/s (48.4MB/s)(7167MiB/155288msec)
   slat (nsec): min=1396, max=398684, avg=4485.68, stdev=3091.75
   clat (usec): min=857, max=148259, avg=8377.47, stdev=8488.15
    lat (usec): min=867, max=148264, avg=8382.02, stdev=8488.16
   clat percentiles (msec):
    |  1.00th=[    3],  5.00th=[    4], 10.00th=[    4], 20.00th=[    4],
    | 30.00th=[    5], 40.00th=[    5], 50.00th=[    5], 60.00th=[    6],
    | 70.00th=[    7], 80.00th=[   13], 90.00th=[   21], 95.00th=[   28],
    | 99.00th=[   42], 99.50th=[   47], 99.90th=[   68], 99.95th=[   68],
    | 99.99th=[  120]
  bw (  KiB/s): min=32224, max=67456, per=99.98%, avg=47254.23, stdev=6792.41, samples=310
  iops        : min= 8056, max=16864, avg=11813.55, stdev=1698.11, samples=310
 lat (usec)   : 250=0.01%, 500=0.04%, 750=0.07%, 1000=0.09%
 lat (msec)   : 2=1.53%, 4=36.85%, 10=41.27%, 20=11.30%, 50=8.61%
 lat (msec)   : 100=0.23%, 250=0.01%
 cpu          : usr=2.13%, sys=9.90%, ctx=925770, majf=0, minf=31
 IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
    submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
    complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
    issued rwts: total=786597,1834843,0,0 short=0,0,0,0 dropped=0,0,0,0
    latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: bw=19.8MiB/s (20.7MB/s), 19.8MiB/s-19.8MiB/s (20.7MB/s-20.7MB/s), io=3073MiB (3222MB), run=155288-155288msec
  WRITE: bw=46.2MiB/s (48.4MB/s), 46.2MiB/s-46.2MiB/s (48.4MB/s-48.4MB/s), io=7167MiB (7516MB), run=155288-155288msec
```

Sequential read IOPS

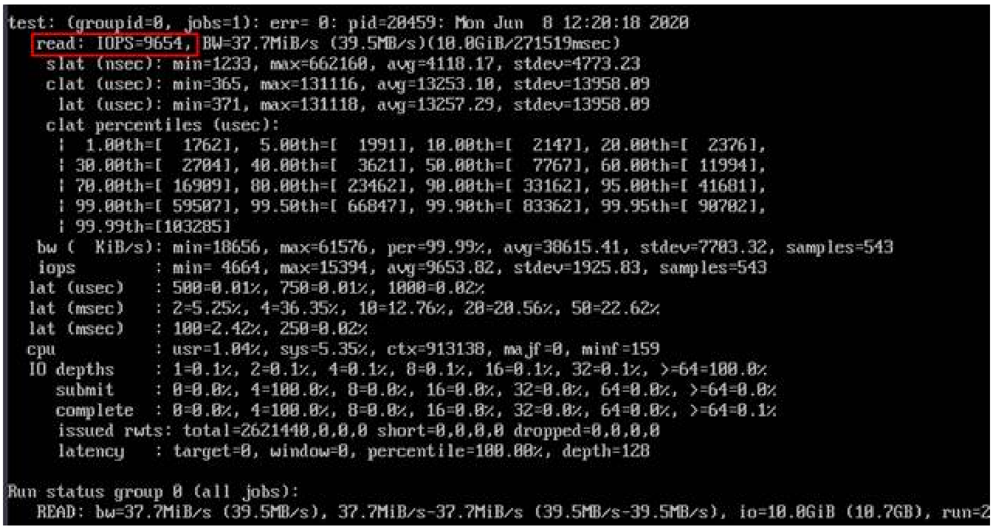
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=read --fallocate=none

  NOTE

  /mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=20459: Mon Jun  8 12:20:18 2020
  read: IOPS=9654, BW=37.7MiB/s (39.5MB/s)(10.0GiB/271519msec)
   slat (nsec): min=1233, max=662160, avg=4118.17, stdev=4773.23
   clat (usec): min=365, max=131116, avg=13253.10, stdev=13958.09
    lat (usec): min=371, max=131118, avg=13257.29, stdev=13958.09
   clat percentiles (usec):
    |  1.00th=[  1762],  5.00th=[  1991], 10.00th=[  2147], 20.00th=[  2376],
    | 30.00th=[  2704], 40.00th=[  3621], 50.00th=[  7767], 60.00th=[ 11994],
    | 70.00th=[ 16909], 80.00th=[ 23462], 90.00th=[ 33162], 95.00th=[ 41681],
    | 99.00th=[ 59507], 99.50th=[ 66847], 99.90th=[ 83362], 99.95th=[ 90702],
    | 99.99th=[103285]
  bw (  KiB/s): min=18656, max=61576, per=99.99%, avg=38615.41, stdev=7703.32, samples=543
  iops        : min= 4664, max=15394, avg=9653.82, stdev=1925.83, samples=543
 lat (usec)   : 500=0.01%, 750=0.01%, 1000=0.02%
 lat (msec)   : 2=5.25%, 4=36.35%, 10=12.76%, 20=20.56%, 50=22.62%
 lat (msec)   : 100=2.42%, 250=0.02%
 cpu          : usr=1.04%, sys=5.35%, ctx=913138, majf=0, minf=159
 IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
    submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
    complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
    issued rwts: total=2621440,0,0,0 short=0,0,0,0 dropped=0,0,0,0
    latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: bw=37.7MiB/s (39.5MB/s), 37.7MiB/s-37.7MiB/s (39.5MB/s-39.5MB/s), io=10.0GiB (10.7GB), run=2
```

Random read IOPS

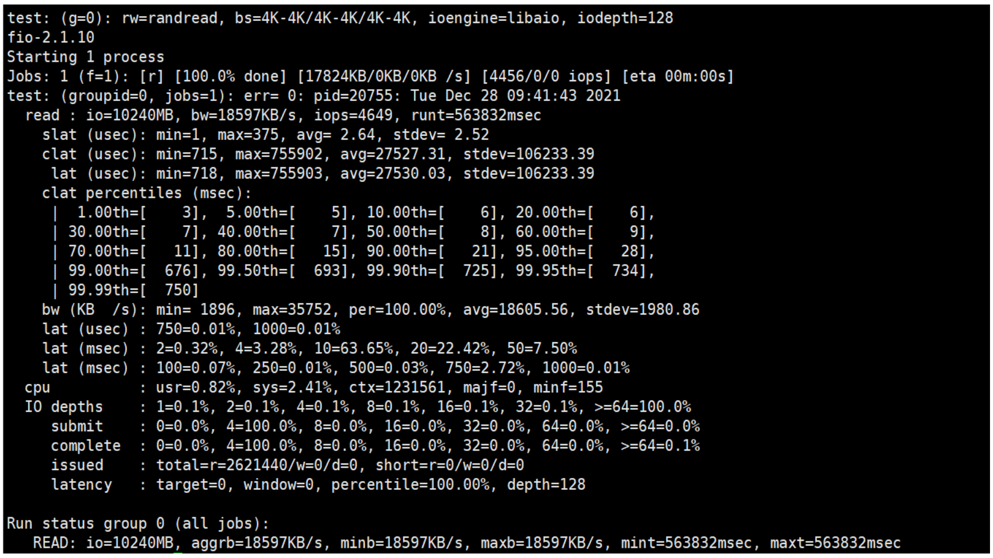
- fio command:

fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=randread --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (g=0): rw=randread, bs=4K-4K/4K-4K/4K-4K, ioengine=libaio, iodepth=128
fio-2.1.10
Starting 1 process
Jobs: 1 (f=1): [r] [100.0% done] [17824KB/0KB/0KB /s] [4456/0/0 iops] [eta 00m:00s]
test: (groupid=0, jobs=1): err= 0: pid=20755: Tue Dec 28 09:41:43 2021
  read : io=10240MB, bw=18597KB/s, iops=4649, runt=563832msec
    slat (usec): min=1, max=375, avg= 2.64, stdev= 2.52
    clat (usec): min=715, max=755902, avg=27527.31, stdev=106233.39
     lat (usec): min=718, max=755903, avg=27530.03, stdev=106233.39
    clat percentiles (msec):
     |  1.00th=[    3],  5.00th=[    5], 10.00th=[    6], 20.00th=[    6],
     | 30.00th=[    7], 40.00th=[    7], 50.00th=[    8], 60.00th=[    9],
     | 70.00th=[   11], 80.00th=[   15], 90.00th=[   21], 95.00th=[   28],
     | 99.00th=[  676], 99.50th=[  693], 99.90th=[  725], 99.95th=[  734],
     | 99.99th=[  750]
    bw (KB  /s): min= 1896, max=35752, per=100.00%, avg=18605.56, stdev=1980.86
    lat (usec) : 750=0.01%, 1000=0.01%
    lat (msec) : 2=0.32%, 4=3.28%, 10=63.65%, 20=22.42%, 50=7.50%
    lat (msec) : 100=0.07%, 250=0.01%, 500=0.03%, 750=2.72%, 1000=0.01%
  cpu          : usr=0.82%, sys=2.41%, ctx=1231561, majf=0, minf=155
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued    : total=r=2621440/w=0/d=0, short=r=0/w=0/d=0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: io=10240MB, aggrb=18597KB/s, minb=18597KB/s, maxb=18597KB/s, mint=563832msec, maxt=563832msec
```

Sequential write IOPS

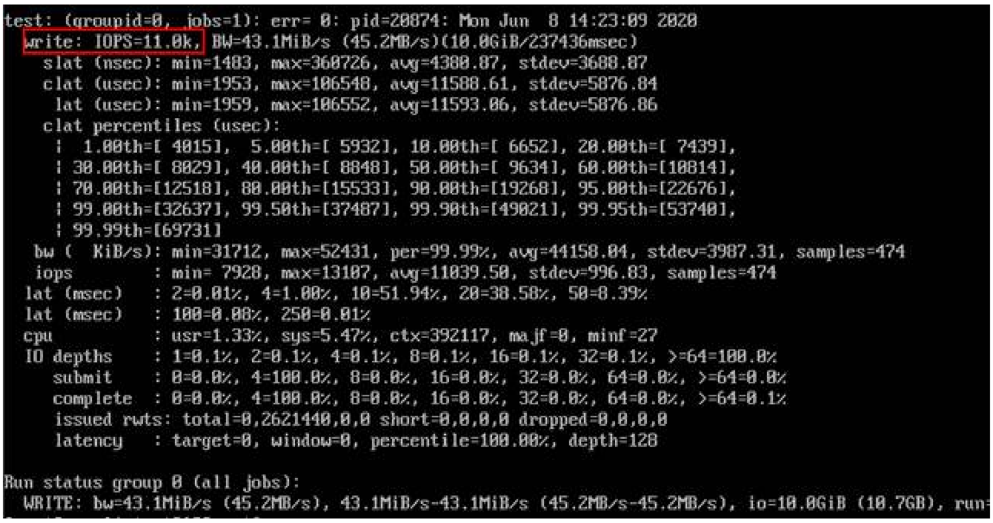
- fio command:

fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=write --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=20874: Mon Jun  8 14:23:09 2020
  write: IOPS=11.0k, BW=43.1MiB/s (45.2MB/s)(10.0GiB/237436msec)
    slat (nsec): min=1483, max=360726, avg=4380.87, stdev=3688.87
    clat (usec): min=1953, max=106548, avg=11588.61, stdev=5876.84
     lat (usec): min=1959, max=106552, avg=11593.06, stdev=5876.86
    clat percentiles (usec):
     |  1.00th=[ 4015],  5.00th=[ 5932], 10.00th=[ 6652], 20.00th=[ 7439],
     | 30.00th=[ 8029], 40.00th=[ 8848], 50.00th=[ 9634], 60.00th=[10814],
     | 70.00th=[12518], 80.00th=[15533], 90.00th=[19268], 95.00th=[22676],
     | 99.00th=[32637], 99.50th=[37487], 99.90th=[49021], 99.95th=[53740],
     | 99.99th=[69731]
   bw (  KiB/s): min=31712, max=52431, per=99.99%, avg=44158.04, stdev=3987.31, samples=474
   iops        : min= 7928, max=13107, avg=11039.50, stdev=996.83, samples=474
  lat (msec)   : 2=0.01%, 4=1.00%, 10=51.94%, 20=38.58%, 50=8.39%
  lat (msec)   : 100=0.08%, 250=0.01%
  cpu          : usr=1.33%, sys=5.47%, ctx=392117, majf=0, minf=27
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued rwts: total=0,2621440,0,0 short=0,0,0,0 dropped=0,0,0,0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
  WRITE: bw=43.1MiB/s (45.2MB/s), 43.1MiB/s-43.1MiB/s (45.2MB/s-45.2MB/s), io=10.0GiB (10.7GB), run=
```

Random write IOPS

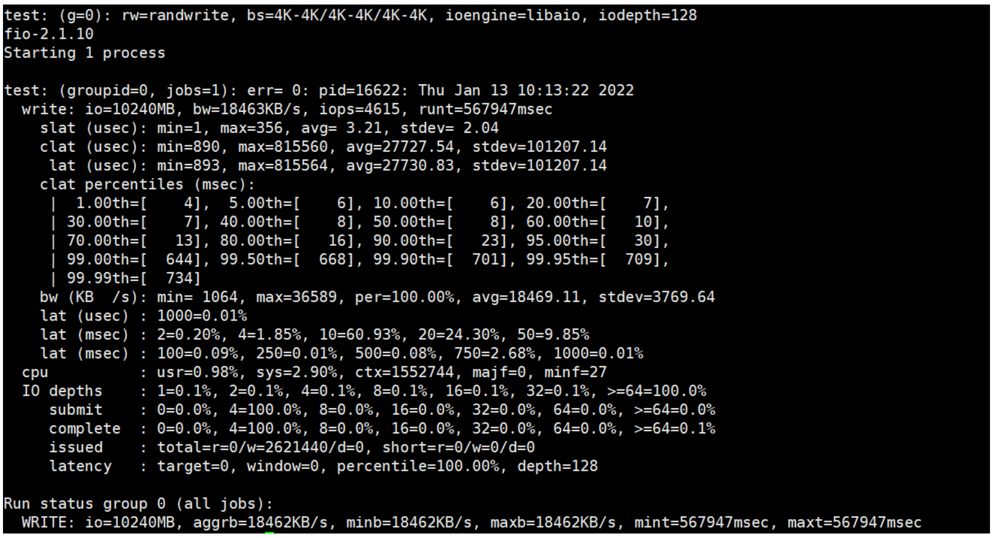
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=4k --iodepth=128 --size=10240M --readwrite=randwrite --fallocate=none

  NOTE

  /mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (g=0): rw=randwrite, bs=4K-4K/4K-4K/4K-4K, ioengine=libaio, iodepth=128
fio-2.1.10
Starting 1 process

test: (groupid=0, jobs=1): err= 0: pid=16622: Thu Jan 13 10:13:22 2022
  write: io=10240MB, bw=18463KB/s, iops=4615, runt=567947msec
    slat (usec): min=1, max=356, avg= 3.21, stdev= 2.04
    clat (usec): min=890, max=815560, avg=27727.54, stdev=101207.14
     lat (usec): min=893, max=815564, avg=27730.83, stdev=101207.14
    clat percentiles (msec):
     |  1.00th=[    4],  5.00th=[    6], 10.00th=[    6], 20.00th=[    7],
     | 30.00th=[    7], 40.00th=[    8], 50.00th=[    8], 60.00th=[   10],
     | 70.00th=[   13], 80.00th=[   16], 90.00th=[   23], 95.00th=[   30],
     | 99.00th=[  644], 99.50th=[  668], 99.90th=[  701], 99.95th=[  709],
     | 99.99th=[  734]
    bw (KB  /s): min= 1064, max=36589, per=100.00%, avg=18469.11, stdev=3769.64
    lat (usec) : 1000=0.01%
    lat (msec) : 2=0.20%, 4=1.85%, 10=60.93%, 20=24.30%, 50=9.85%
    lat (msec) : 100=0.09%, 250=0.01%, 500=0.08%, 750=2.68%, 1000=0.01%
  cpu          : usr=0.98%, sys=2.90%, ctx=1552744, majf=0, minf=27
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=0.1%, >=64=100.0%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued    : total=r=0/w=2621440/d=0, short=r=0/w=0/d=0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
  WRITE: io=10240MB, aggrb=18462KB/s, minb=18462KB/s, maxb=18462KB/s, mint=567947msec, maxt=567947msec
```

Sequential read bandwidth

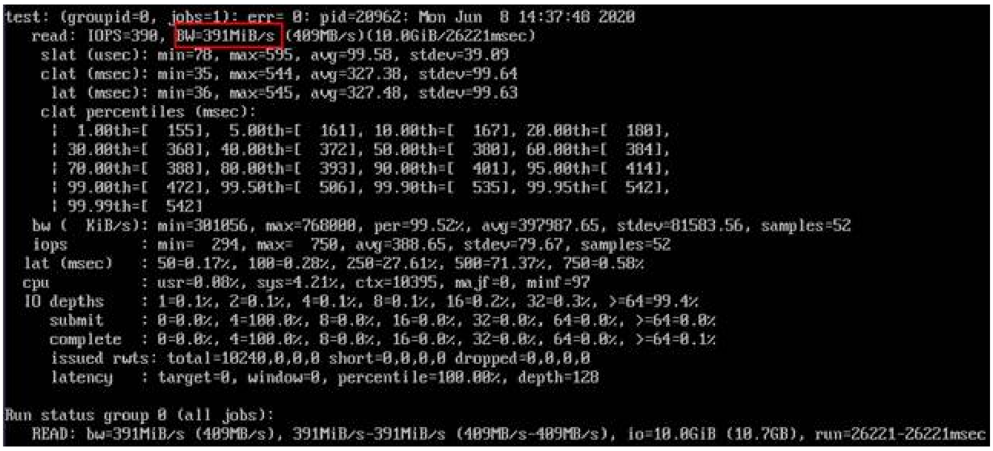
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=1M --iodepth=128 --size=10240M --readwrite=read --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=20962: Mon Jun  8 14:37:48 2020
   read: IOPS=390, BW=391MiB/s (409MB/s)(10.0GiB/26221msec)
    slat (usec): min=78, max=595, avg=99.58, stdev=39.09
    clat (msec): min=35, max=544, avg=327.38, stdev=99.64
     lat (msec): min=36, max=545, avg=327.48, stdev=99.63
    clat percentiles (msec):
     |  1.00th=[  155],  5.00th=[  161], 10.00th=[  167], 20.00th=[  180],
     | 30.00th=[  368], 40.00th=[  372], 50.00th=[  380], 60.00th=[  384],
     | 70.00th=[  388], 80.00th=[  393], 90.00th=[  401], 95.00th=[  414],
     | 99.00th=[  472], 99.50th=[  506], 99.90th=[  535], 99.95th=[  542],
     | 99.99th=[  542]
   bw (  KiB/s): min=301056, max=768000, per=99.52%, avg=397987.65, stdev=81583.56, samples=52
   iops        : min=  294, max=  750, avg=388.65, stdev=79.67, samples=52
  lat (msec)   : 50=0.17%, 100=0.28%, 250=27.61%, 500=71.37%, 750=0.58%
  cpu          : usr=0.00%, sys=4.21%, ctx=10395, majf=0, minf=97
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.2%, 32=0.3%, >=64=99.4%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued rwts: total=10240,0,0,0 short=0,0,0,0 dropped=0,0,0,0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: bw=391MiB/s (409MB/s), 391MiB/s-391MiB/s (409MB/s-409MB/s), io=10.0GiB (10.7GB), run=26221-26221msec
```

Random read bandwidth

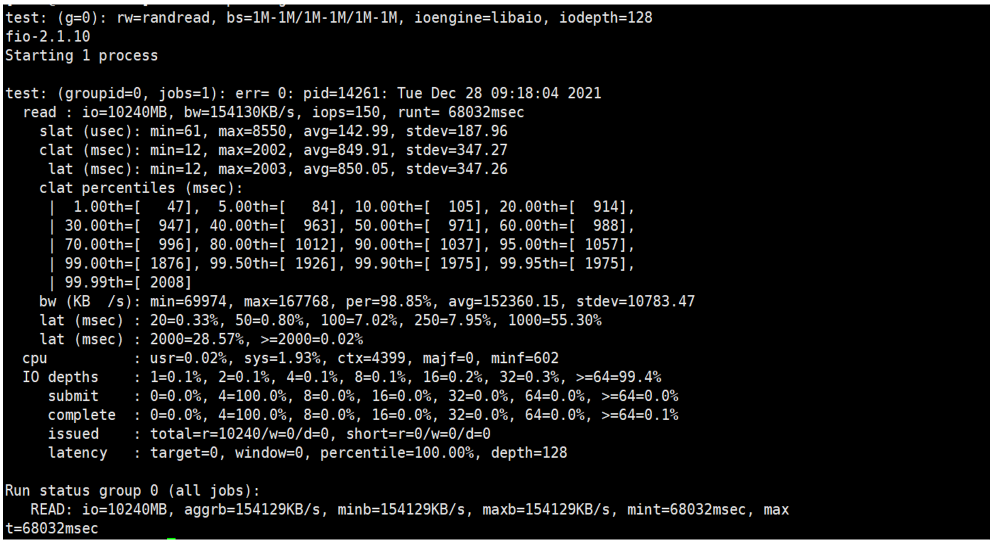
- fio command:

  fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=1M --iodepth=128 --size=10240M --readwrite=randread --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (g=0): rw=randread, bs=1M-1M/1M-1M/1M-1M, ioengine=libaio, iodepth=128
fio-2.1.10
Starting 1 process

test: (groupid=0, jobs=1): err= 0: pid=14261: Tue Dec 28 09:18:04 2021
  read : io=10240MB, bw=154130KB/s, iops=150, runt= 68032msec
    slat (usec): min=61, max=8550, avg=142.99, stdev=187.96
    clat (msec): min=12, max=2002, avg=849.91, stdev=347.27
     lat (msec): min=12, max=2003, avg=850.05, stdev=347.26
    clat percentiles (msec):
     |  1.00th=[   47],  5.00th=[   84], 10.00th=[  105], 20.00th=[  914],
     | 30.00th=[  947], 40.00th=[  963], 50.00th=[  971], 60.00th=[  988],
     | 70.00th=[  996], 80.00th=[ 1012], 90.00th=[ 1037], 95.00th=[ 1057],
     | 99.00th=[ 1876], 99.50th=[ 1926], 99.90th=[ 1975], 99.95th=[ 1975],
     | 99.99th=[ 2008]
    bw (KB  /s): min=69974, max=167768, per=98.85%, avg=152360.15, stdev=10783.47
    lat (msec) : 20=0.33%, 50=0.80%, 100=7.02%, 250=7.95%, 1000=55.30%
    lat (msec) : 2000=28.57%, >=2000=0.02%
  cpu          : usr=0.02%, sys=1.93%, ctx=4399, majf=0, minf=602
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.2%, 32=0.3%, >=64=99.4%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued    : total=r=10240/w=0/d=0, short=r=0/w=0/d=0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
   READ: io=10240MB, aggrb=154129KB/s, minb=154129KB/s, maxb=154129KB/s, mint=68032msec, max
t=68032msec
```

Sequential write bandwidth

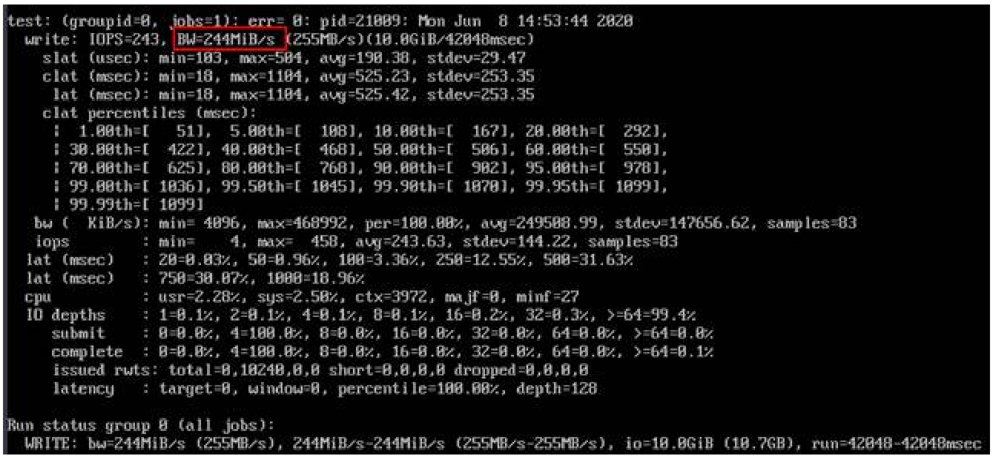
- fio command:

fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=1M --iodepth=128 --size=10240M --readwrite=write --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (groupid=0, jobs=1): err= 0: pid=21009: Mon Jun  8 14:53:44 2020
  write: IOPS=243, BW=244MiB/s (255MB/s)(10.0GiB/42048msec)
    slat (usec): min=103, max=504, avg=190.38, stdev=29.47
    clat (msec): min=18, max=1104, avg=525.23, stdev=253.35
     lat (msec): min=18, max=1104, avg=525.42, stdev=253.35
    clat percentiles (msec):
     |  1.00th=[   51],  5.00th=[  108], 10.00th=[  167], 20.00th=[  292],
     | 30.00th=[  422], 40.00th=[  468], 50.00th=[  506], 60.00th=[  550],
     | 70.00th=[  625], 80.00th=[  768], 90.00th=[  902], 95.00th=[  978],
     | 99.00th=[ 1036], 99.50th=[ 1045], 99.90th=[ 1070], 99.95th=[ 1099],
     | 99.99th=[ 1099]
   bw (  KiB/s): min= 4096, max=468992, per=100.00%, avg=249508.99, stdev=147656.62, samples=83
   iops        : min=    4, max=  458, avg=243.63, stdev=144.22, samples=83
  lat (msec)   : 20=0.03%, 50=0.96%, 100=3.36%, 250=12.55%, 500=31.63%
  lat (msec)   : 750=30.07%, 1000=18.96%
  cpu          : usr=2.28%, sys=2.50%, ctx=3972, majf=0, minf=27
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.2%, 32=0.3%, >=64=99.4%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued rwts: total=0,10240,0,0 short=0,0,0,0 dropped=0,0,0,0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
  WRITE: bw=244MiB/s (255MB/s), 244MiB/s-244MiB/s (255MB/s-255MB/s), io=10.0GiB (10.7GB), run=42048-42048msec
```

Random write bandwidth

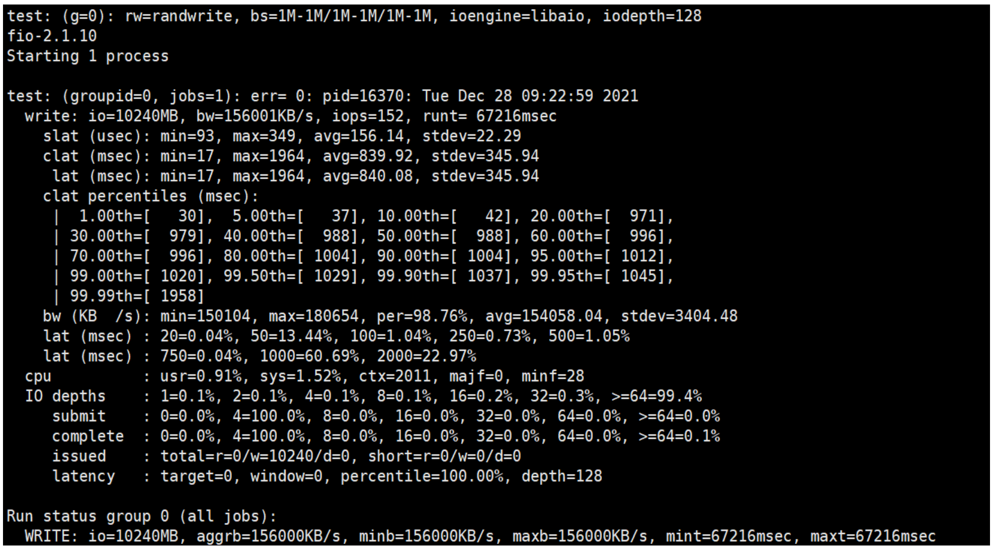
- fio command:

fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=*/mnt/sfs-turbo/test_fio* --bs=1M --iodepth=128 --size=10240M --readwrite=randwrite --fallocate=none

NOTE

/mnt/sfs-turbo/test_fio indicates the location of the file to be tested. The location must be specific to the file name, which is the **test_fio** file in the **/mnt/sfs-turbo** directory in this example. Set it based on the site requirements.

- fio result:



```
test: (g=0): rw=randwrite, bs=1M-1M/1M-1M/1M-1M, ioengine=libaio, iodepth=128
fio-2.1.10
Starting 1 process

test: (groupid=0, jobs=1): err= 0: pid=16370: Tue Dec 28 09:22:59 2021
  write: io=10240MB, bw=156001KB/s, iops=152, runt= 67216msec
    slat (usec): min=93, max=349, avg=156.14, stdev=22.29
    clat (msec): min=17, max=1964, avg=839.92, stdev=345.94
     lat (msec): min=17, max=1964, avg=840.08, stdev=345.94
    clat percentiles (msec):
     |  1.00th=[   30],  5.00th=[   37], 10.00th=[   42], 20.00th=[  971],
     | 30.00th=[  979], 40.00th=[  988], 50.00th=[  988], 60.00th=[  996],
     | 70.00th=[  996], 80.00th=[ 1004], 90.00th=[ 1004], 95.00th=[ 1012],
     | 99.00th=[ 1020], 99.50th=[ 1029], 99.90th=[ 1037], 99.95th=[ 1045],
     | 99.99th=[ 1958]
    bw (KB  /s): min=150104, max=180654, per=98.76%, avg=154058.04, stdev=3404.48
    lat (msec) : 20=0.04%, 50=13.44%, 100=1.04%, 250=0.73%, 500=1.05%
    lat (msec) : 750=0.04%, 1000=60.69%, 2000=22.97%
  cpu          : usr=0.91%, sys=1.52%, ctx=2011, majf=0, minf=28
  IO depths    : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.2%, 32=0.3%, >=64=99.4%
     submit    : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
     complete  : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.1%
     issued    : total=r=0/w=10240/d=0, short=r=0/w=0/d=0
     latency   : target=0, window=0, percentile=100.00%, depth=128

Run status group 0 (all jobs):
  WRITE: io=10240MB, aggrb=156000KB/s, minb=156000KB/s, maxb=156000KB/s, mint=67216msec, maxt=67216msec
```

7.2 Mounting a File System to a Linux ECS as a Non-root User

Scenarios

By default, a Linux ECS allows only the **root** user to run the **mount** command for mounting a file system. However, if the permissions of user **root** are assigned to other common users, such users can also run the **mount** command for file system mounting. The following describes how to mount a file system to a Linux ECS as a common user. The EulerOS is used as an example.

Prerequisites

- A non-**root** user has been created on the ECS.
- A file system has been created and can be mounted to the ECS by the **root** user.
- You have obtained the mount point of the file system.

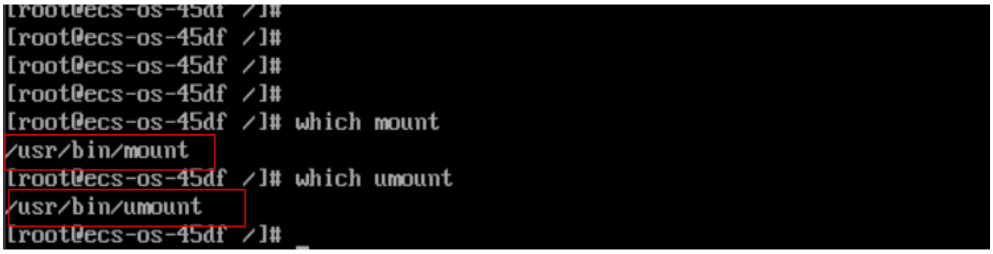
Procedure

Step 1 Log in to the ECS as user **root**.

Step 2 Assign the permissions of user **root** to the non-**root** user.

1. Run the **chmod 777 /etc/sudoers** command to change the **sudoers** file to be editable.
2. Use the **which** command to view the **mount** and **umount** command paths.

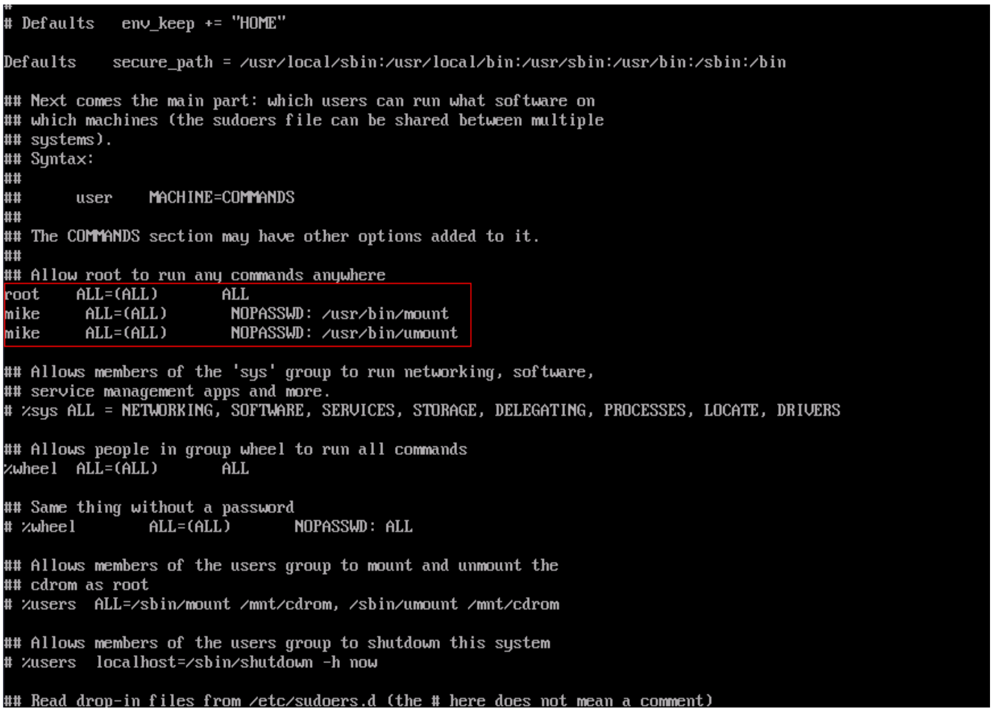
Figure 7-1 Viewing command paths



```
[root@ecs-os-45df /]#
[root@ecs-os-45df /]#
[root@ecs-os-45df /]#
[root@ecs-os-45df /]#
[root@ecs-os-45df /]# which mount
/usr/bin/mount
[root@ecs-os-45df /]# which umount
/usr/bin/umount
[root@ecs-os-45df /]# _
```

3. Run the **vi /etc/resolv.conf** command to edit the **sudoers** file.
4. Add a common user under the **root** account. In the following figure, user **Mike** is added.

Figure 7-2 Adding a user



```
# Defaults   env_keep += "HOME"

Defaults    secure_path = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
mike     ALL=(ALL)       NOPASSWD: /usr/bin/mount
mike     ALL=(ALL)       NOPASSWD: /usr/bin/umount

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)       ALL

## Same thing without a password
# %wheel        ALL=(ALL)       NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
```

5. Press **Esc**, input **:wq**, and press **Enter** to save and exit.
6. Run the **chmod 440 /etc/sudoers** command to change the **sudoers** file to be read-only.

Step 3 Log in to the ECS as user **Mike**.

Step 4 Run the following command to mount the file system. For details about the mounting parameters, see **Table 7-2**.

sudo mount -t nfs -o vers=3,timeo=600,noresvport,nolock *Mount point Local path*

Table 7-2 Parameter description

Parameter	Description
Mount point	The format of an SFS Capacity-Oriented file system is *File system domain name*:/*Path*, for example, **example.com:/share-***xxx*. The format of an SFS Turbo file system is *File system IP address*:/, for example, **192.168.0.0:/**. **NOTE** *x* is a digit or letter. If the mount point is too long to display completely, you can adjust the column width.
Local path	Local path on the ECS, used to mount the file system, for example, **/local_path**.

Step 5 Run the following command to view the mounted file system:

mount -l

If the command output contains the following information, the file system has been mounted.

```
example.com:/share-xxx on /local_path type nfs (rw,vers=3,timeo=600,nolock,addr=)
```

----End

7.3 Mounting a Subdirectory of an NFS File System to ECSs (Linux)

This section describes how to mount a subdirectory of an NFS file system to Linux ECSs.

Prerequisites

You have mounted a file system to Linux ECSs by referring to **Mounting an NFS File System to ECSs (Linux)**.

Procedure

Step 1 Run the following command to create a subdirectory in the local path:

mkdir *Local path*/*Subdirectory*

NOTE

Variable *Local path* is an ECS local directory where the file system will be mounted on, for example, **/local_path**. Specify the local path used for mounting the root directory.

Step 2 Run the following command to mount the subdirectory to the ECSs that are in the same VPC as the file system: (Currently, the file system can be mounted to Linux ECSs using NFS v3 only.)

mount -t nfs -o vers=3,timeo=600,noresvport,nolock *Domain name or IP address of the file system*:**/***Subdirectory Local path*

NOTE

- *Domain name or IP address of the file system*: You can obtain it in the file system list from the console.
  - SFS Capacity-Oriented: *example.com:/share-xxx/subdirectory*
  - SFS Turbo: *xx.xx.xx.xx:/subdirectory*
- *Subdirectory*: Specify the subdirectory created in the previous step.
- *Local path*: An ECS local directory where the file system is mounted, for example, **/local_path**. Specify the local path used for mounting the root directory.

Step 3 Run the following command to view the mounted file system:

mount -l

If the command output contains the following information, the file system has been mounted.

```
Mount point on /local_path type nfs (rw,vers=3,timeo=600,nolock,addr=)
```

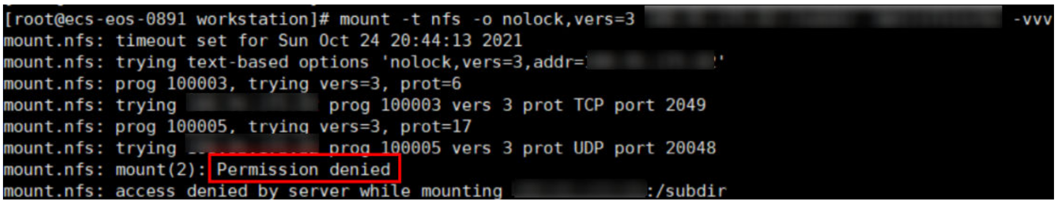
Step 4 After the subdirectory has been mounted, you can access it from the server, and read or write data.

----End

Troubleshooting

If a subdirectory is not created before mounting, the mounting will fail.

Figure 7-3 Mounting without a subdirectory created

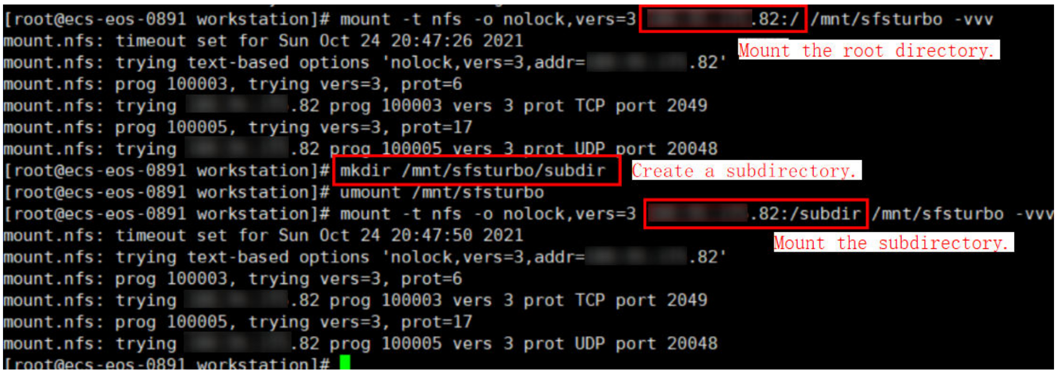


```
[root@ecs-eos-0891 workstation]# mount -t nfs -o nolock,vers=3 [redacted] -vvv
mount.nfs: timeout set for Sun Oct 24 20:44:13 2021
mount.nfs: trying text-based options 'nolock,vers=3,addr=[redacted]'
mount.nfs: prog 100003, trying vers=3, prot=6
mount.nfs: trying [redacted] prog 100003 vers 3 prot TCP port 2049
mount.nfs: prog 100005, trying vers=3, prot=17
mount.nfs: trying [redacted] prog 100005 vers 3 prot UDP port 20048
mount.nfs: mount(2): Permission denied
mount.nfs: access denied by server while mounting [redacted]:/subdir
```

In the preceding figure, the root directory does not have the **subdir** subdirectory created so that the mounting fails. In this case, error message "Permission denied" is reported.

To troubleshoot this issue, mount the root directory, create a subdirectory, and then mount the subdirectory.

Figure 7-4 Mounting subdirectory



```
[root@ecs-eos-0891 workstation]# mount -t nfs -o nolock,vers=3 [redacted].82:/ /mnt/sfsturbo -vvv
mount.nfs: timeout set for Sun Oct 24 20:47:26 2021
mount.nfs: trying text-based options 'nolock,vers=3,addr=[redacted].82'
mount.nfs: prog 100003, trying vers=3, prot=6
mount.nfs: trying [redacted].82 prog 100003 vers 3 prot TCP port 2049
mount.nfs: prog 100005, trying vers=3, prot=17
mount.nfs: trying [redacted].82 prog 100005 vers 3 prot UDP port 20048
[root@ecs-eos-0891 workstation]# mkdir /mnt/sfsturbo/subdir
[root@ecs-eos-0891 workstation]# umount /mnt/sfsturbo
[root@ecs-eos-0891 workstation]# mount -t nfs -o nolock,vers=3 [redacted].82:/subdir /mnt/sfsturbo -vvv
mount.nfs: timeout set for Sun Oct 24 20:47:50 2021
mount.nfs: trying text-based options 'nolock,vers=3,addr=[redacted].82'
mount.nfs: prog 100003, trying vers=3, prot=6
mount.nfs: trying [redacted].82 prog 100003 vers 3 prot TCP port 2049
mount.nfs: prog 100005, trying vers=3, prot=17
mount.nfs: trying [redacted].82 prog 100005 vers 3 prot UDP port 20048
[root@ecs-eos-0891 workstation]#
```

Mount the root directory.

Create a subdirectory.

Mount the subdirectory.

7.4 Data Migration

7.4.1 Migration Description

By default, an SFS Turbo file system can only be accessed by ECSs or CCE contains that reside in the same VPC as the file system. To access an SFS Turbo file system from an on-premises data center or a different VPC, you need to establish network connections by using Direct Connect, VPN, or VPC peering connections.

- Access from on premises or another cloud: Use Direct Connect or VPN.
- On-cloud, cross-VPC access using the same account in a given region: Use VPC peering.
- On-cloud, cross-account access in a given region: Use VPC peering.

- On-cloud, cross-region access: Use Cloud Connect.

You can migrate data to SFS Turbo using an ECS that can access the Internet.

- Mount the SFS Turbo file system to the ECS and migrate data from the local NAS storage to the SFS Turbo file system.

  Using Direct Connect to Migrate Data
- If communication cannot be enabled through file system mounting, migrate data using the ECS via the Internet.

  Using the Internet to Migrate Data

7.4.2 Using Direct Connect to Migrate Data

Context

You can migrate data from a local NAS to SFS Turbo using Direct Connect.

In this solution, a Linux ECS is created to connect the local NAS and SFS Turbo, and data is migrated to the cloud using an ECS.

You can also refer to this solution to migrate data from an on-cloud NAS to SFS Turbo. For details, see **Migrating Data from On-cloud NAS to SFS**.

Limitations and Constraints

- Only Linux ECSs can be used to migrate data.
- The UID and GID of your file will no longer be consistent after data migration.
- The file access modes will no longer be consistent after data migration.
- Incremental migration is supported, so that only changed data is migrated.

Prerequisites

- You have enabled and configured Direct Connect. For details, see *Direct Connect User Guide*.
- You have created a Linux ECS.
- You have created an SFS Turbo file system and have obtained the mount point of the file system.
- You have obtained the mount point of the local NAS.

Procedure

Step 1 Log in to the ECS console.

Step 2 Log in to the created Linux ECS to access the local NAS and SFS Turbo file system.

Step 3 Run the following mount command to access the local NAS:

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock Mount point of the local NAS /mnt/src
```

Step 4 Run the following mount command to access the file system:

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock Mount point of the file system /mnt/dst
```

Step 5 Run the following commands on the Linux ECS to install the rclone tool:

```
wget https://downloads.rclone.org/v1.53.4/rclone-v1.53.4-linux-amd64.zip --no-check-certificate
unzip rclone-v1.53.4-linux-amd64.zip
chmod 0755 ./rclone-*/rclone
cp ./rclone-*/rclone /usr/bin/
rm -rf ./rclone-*
```

Step 6 Run the following command to synchronize data:

```
rclone copy /mnt/src /mnt/dst -P --transfers 32 --checkers 64 --links --create-empty-src-dirs
```

NOTE

> Set **transfers** and **checkers** based on the system specifications. The parameters are described as follows:
>
> - **--transfers**: number of files that can be transferred concurrently
> - **--checkers**: number of local files that can be scanned concurrently
> - **-P**: data copy progress
> - **--links**: replicates the soft links from the source. They are saved as soft links in the destination.
>
>   **--copy-links**: replicates the content of files to which the soft links point. They are saved as files rather than soft links in the destination.
> - **--create-empty-src-dirs**: replicates the empty directories from the source to the destination.

After data synchronization is complete, go to the target file system to check whether data is migrated.

----End

Migrating Data from On-cloud NAS to SFS

To migrate data from an on-cloud NAS to your SFS Turbo file system, ensure that the NAS and file system are in the same VPC, or you can use Cloud Connect to migrate data.

For details about how to configure Cloud Connect, see *Direct Connect User Guide*.

7.4.3 Using the Internet to Migrate Data

Context

You can migrate data from a local NAS to SFS Turbo using the Internet.

In this solution, to migrate data from the local NAS to the cloud, a Linux server is created both on the cloud and on-premises. Inbound and outbound traffic is allowed on port 22 of these two servers. The on-premises server is used to access the local NAS, and the ECS is used to access SFS Turbo.

You can also refer to this solution to migrate data from an on-cloud NAS to SFS Turbo.

Limitations and Constraints

- Data cannot be migrated from the local NAS to SFS Capacity-Oriented using the Internet.
- Only Linux ECSs can be used to migrate data.
- The UID and GID of your file will no longer be consistent after data migration.

- The file access modes will no longer be consistent after data migration.
- Inbound and outbound traffic must be allowed on port 22.
- Incremental migration is supported, so that only changed data is migrated.

Prerequisites

- A Linux server has been created on the cloud and on-premises respectively.
- EIPs have been configured for the servers to ensure that the two servers can communicate with each other.
- You have created an SFS Turbo file system and have obtained the mount point of the file system.
- You have obtained the mount point of the local NAS.

Procedure

Step 1 Log in to the ECS console.

Step 2 Log in to the created on-premises server **client1** and run the following command to access the local NAS:

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock Mount point of the local NAS /mnt/src
```

Step 3 Log in to the created Linux ECS **client2** and run the following command to access the SFS Turbo file system:

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock Mount point of the SFS Turbo file system  /mnt/dst
```

Step 4 Run the following commands on **client1** to install the rclone tool:

```
wget https://downloads.rclone.org/v1.53.4/rclone-v1.53.4-linux-amd64.zip --no-check-certificate
unzip rclone-v1.53.4-linux-amd64.zip
chmod 0755 ./rclone-*/rclone
cp ./rclone-*/rclone /usr/bin/
rm -rf ./rclone-*
```

Step 5 Run the following commands on **client1** to configure the environment:

```
rclone config
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
name> remote name (New name)
Type of storage to configure.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value
24 / SSH/SFTP Connection
   \ "sftp"
Storage> 24 (Select the SSH/SFTP number)
SSH host to connect to
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value
 1 / Connect to example.com
   \ "example.com"
host> ip address (IP address of client2)
SSH username, leave blank for current username, root
Enter a string value. Press Enter for the default ("").
user> user name (Username of client2)
SSH port, leave blank to use default (22)
Enter a string value. Press Enter for the default ("").
port> 22
SSH password, leave blank to use ssh-agent.
y) Yes type in my own password
g) Generate random password
```

```
n) No leave this optional password blank
y/g/n> y
Enter the password:
password: (Password for logging in to client2)
Confirm the password:
password: (Confirm the password for logging in to client2)
Path to PEM-encoded private key file, leave blank or set key-use-agent to use ssh-agent.
Enter a string value. Press Enter for the default ("").
key_file> (Press Enter)
The passphrase to decrypt the PEM-encoded private key file.

Only PEM encrypted key files (old OpenSSH format) are supported. Encrypted keys
in the new OpenSSH format can't be used.
y) Yes type in my own password
g) Generate random password
n) No leave this optional password blank
y/g/n> n
When set forces the usage of the ssh-agent.
When key-file is also set, the ".pub" file of the specified key-file is read and only the associated key is
requested from the ssh-agent. This allows to avoid `Too many authentication failures for *username*` errors
when the ssh-agent contains many keys.
Enter a boolean value (true or false). Press Enter for the default ("false").
key_use_agent> (Press Enter)
Enable the use of the aes128-cbc cipher. This cipher is insecure and may allow plaintext data to be
recovered by an attacker.
Enter a boolean value (true or false). Press Enter for the default ("false").
Choose a number from below, or type in your own value
 1 / Use default Cipher list.
   \ "false"
 2 / Enables the use of the aes128-cbc cipher.
   \ "true"
use_insecure_cipher> (Press Enter)
Disable the execution of SSH commands to determine if remote file hashing is available.
Leave blank or set to false to enable hashing (recommended), set to true to disable hashing.
Enter a boolean value (true or false). Press Enter for the default ("false").
disable_hashcheck>
Edit advanced config? (y/n)
y) Yes
n) No
y/n> n
Remote config
--------------------
[remote_name]
type = sftp
host=(client2 ip)
user=(client2 user name)
port = 22
pass = *** ENCRYPTED ***
key_file_pass = *** ENCRYPTED ***
--------------------
y) Yes this is OK
e) Edit this remote
d) Delete this remote
y/e/d> y
Current remotes:

Name                 Type
====                 ====
remote_name          sftp

e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q> q
```

Step 6 Run the following command to view the **rclone.conf** file in **/root/.config/rclone/rclone.conf**:

```
cat /root/.config/rclone/rclone.conf
[remote_name]
type = sftp
host=(client2 ip)
user=(client2 user name)
port = 22
pass = ***
key_file_pass = ***
```

Step 7 Run the following command on **client1** to synchronize data:

```
rclone copy /mnt/src remote_name:/mnt/dst -P --transfers 32 --checkers 64
```

NOTE

- Replace *remote_name* in the command with the remote name in the environment.
- Set **transfers** and **checkers** based on the system specifications. The parameters are described as follows:
  - **transfers**: number of files that can be transferred concurrently
  - **checkers**: number of local files that can be scanned concurrently
  - **P**: data copy progress

After data synchronization is complete, go to the SFS Turbo file system to check whether data is migrated.

----End

7.4.4 Migrating Data Between File Systems

Solution Overview

You can migrate data from an SFS Capacity-Oriented file system to an SFS Turbo file system or the other way around.

This solution creates a Linux ECS to connect an SFS Capacity-Oriented file system with an SFS Turbo file system.

Limitations and Constraints

- Only Linux ECSs can be used to migrate data.
- The Linux ECS, SFS Capacity-Oriented file system, and SFS Turbo file system must be in the same VPC.
- Incremental migration is supported, so that only changed data is migrated.

Prerequisites

- You have created a Linux ECS.
- You have created an SFS Capacity-Oriented file system and an SFS Turbo file system and have obtained their mount points.

Procedure

Step 1 Log in to the ECS console.

Step 2 Log in to the created Linux ECS that can access SFS Capacity-Oriented and SFS Turbo file systems.

Step 3 Run the following command to mount file system 1 (either the SFS Capacity-Oriented or SFS Turbo file system). After that, you can access file system 1 on the Linux ECS.

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock [Mount point of file system 1] /mnt/src
```

Step 4 Run the following command to mount file system 2 (the other file system that you have not mounted in the previous step). After that, you can access file system 2 on the Linux ECS.

```
mount -t nfs -o vers=3,timeo=600,noresvport,nolock [Mount point of file system 2] /mnt/dst
```

Step 5 Run the following commands on the Linux ECS to install the rclone tool:

```
wget https://downloads.rclone.org/v1.53.4/rclone-v1.53.4-linux-amd64.zip --no-check-certificate
unzip rclone-v1.53.4-linux-amd64.zip
chmod 0755 ./rclone-*/rclone
cp ./rclone-*/rclone /usr/bin/
rm -rf ./rclone-*
```

Step 6 Run the following command to synchronize data:

```
rclone copy /mnt/src /mnt/dst -P --transfers 32 --checkers 64 --links --create-empty-src-dirs
```

NOTE

Set **transfers** and **checkers** based on the system specifications. The parameters are described as follows:

- **--transfers**: number of files that can be transferred concurrently
- **--checkers**: number of local files that can be scanned concurrently
- **-P**: data copy progress
- **--links**: replicates the soft links from the source. They are saved as soft links in the destination.

  --copy-links: replicates the content of files to which the soft links point. They are saved as files rather than soft links in the destination.
- **--create-empty-src-dirs**: replicates the empty directories from the source to the destination.

After data synchronization is complete, go to the target file system to check whether data is migrated.

----End

Verification

Step 1 Log in to the created Linux ECS.

Step 2 Run the following commands on the destination server to verify file synchronization:

```
cd /mnt/dst
ls | wc -l
```

Step 3 If the data volume is the same as that on the source server, the data is migrated successfully.

----End

A Change History

Released On	Description
2023-09-20	This issue is the fifth official release, which incorporates the following changes: ● Service Overview: – Added section **Dedicated SFS Turbo**. – Added the descriptions about encryption and CTS in section **SFS and Other Services**. ● Management: – Added section **Creating a Custom Policy**. – Added section **Backup**. ● Troubleshooting: – Added section **File System Performance Is Poor**. ● FAQs: – Concepts: Added **What Is SFS Turbo?**. – Restrictions: Added **Can I Migrate My File System Data to Another Region?** and **Can a File System Be Mounted to Multiple Accounts?** – Networks: Added **Does SFS Support Cross-Region Mounting?** and **What Can I Do If the Data of the File System That Is Mounted to Two Servers Is Not Synchronized?** – Others: Added **Why Is the Capacity Displayed as 10P After I Mount My SFS Capacity-Oriented File System?**, **Can a File System Be Accessed Across Multiple AZs?**, **Can I Upgrade an SFS Capacity-Oriented File System to an SFS Turbo File System?**, **Can I Upgrade an SFS Turbo File System from Standard to Standard-Enhanced?**, **How Can I Migrate Data Between SFS and EVS?**, **Can I Directly Access SFS from On-premises Devices?**, **How Do I Delete .nfs Files?**, **Why My File System Used Space**

Released On	Description
	Increases After I Migrate from SFS Capacity-Oriented to SFS Turbo?, **How Can I Improve the Copy and Delete Efficiency with an SFS Turbo File System?**, **How Do Second- and Third-level Directory Permissions of an SFS Turbo File System Be Inherited?**, and **How Do I Deploy SFS Turbo on CCE?** ● Other Operations: – Added section **Testing SFS Turbo Performance**. – Added section **Mounting a Subdirectory of an NFS File System to ECSs (Linux)**. – Data Migration: Added sections **Migration Description** and **Migrating Data Between File Systems**.
2021-09-30	This issue is the fourth official release, which incorporates the following change: Added descriptions about permissions management.
2021-04-17	This issue is the third official release, which incorporates the following change: Added descriptions that SFS Turbo supports encryption.
2021-01-20	This issue is the second official release, which incorporates the following change: This is the first time that SFS Turbo file systems go online. Required notes are added accordingly.
2020-02-26	This issue is the first official release.