

Scalable File Service

User Guide

Issue 08
Date 2024-06-26



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

1 Permissions Management.....	1
1.1 Creating a User and Granting SFS Permissions.....	1
1.2 Creating a Custom Policy.....	3
2 File System Management.....	5
2.1 Viewing a File System.....	5
2.2 Deleting a File System.....	6
3 Network Configuration.....	9
3.1 Configuring Multi-VPC Access.....	9
3.2 Configuring Multi-Account Access.....	16
3.3 Configuring DNS.....	21
4 File System Resizing.....	25
5 Quotas.....	29
6 Encryption.....	31
7 Backup.....	32
8 Monitoring.....	35
8.1 SFS Metrics.....	35
8.2 SFS Turbo Metrics.....	38
8.3 Creating Alarm Rules.....	40
9 Auditing.....	43
9.1 Supported SFS Operations.....	43
10 Typical Applications.....	46
10.1 High-performance Computing.....	46
10.2 Media Processing.....	48
10.3 Enterprise Website/App Background.....	49
10.4 Log Printing.....	50
11 Other Operations.....	52
11.1 Testing SFS Turbo Performance.....	52
11.2 Mounting a File System to a Linux ECS as a Non-root User.....	58
11.3 Mounting a Subdirectory of an NFS File System to ECSs (Linux).....	61

11.4 Data Migration.....	62
11.4.1 Migration Description.....	62
11.4.2 Using Direct Connect to Migrate Data.....	63
11.4.3 Using the Internet to Migrate Data.....	64
11.4.4 Migrating Data Between File Systems.....	67

1 Permissions Management

1.1 Creating a User and Granting SFS Permissions

This section 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 Huawei Cloud account does not require individual IAM users, skip this section.

This section describes the procedure for granting permissions (see [Figure 1-1](#)).

Prerequisites

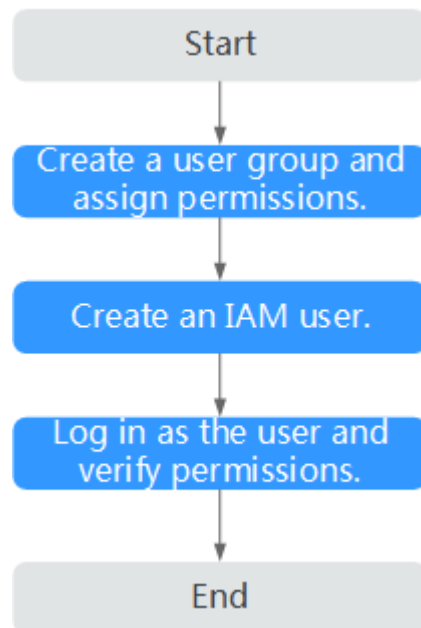
Learn about the permissions (see [System-defined roles and policies](#)) supported by SFS and choose policies or roles according to your requirements. For the permissions of other services, see [System Permissions](#).

Use 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 for SFS Turbo file systems.
- Only system-defined policies are supported in General Purpose file systems. Custom policies are not supported.

Process Flow

Figure 1-1 Process for granting SFS permissions



1. **Create a user group and assign permissions** to it.
On the IAM console, create a user group and grant it read-only permissions:
For SFS Capacity-Oriented, grant the **SFS ReadOnlyAccess** policy.
For SFS Turbo, grant the **SFS Turbo ReadOnlyAccess** policy.
2. **Create a user** and add it to a user group.
Create a user on the IAM console and add the user to the group created in 1.
3. **Log in** and verify permissions.
Log in to the SFS console using the created user, and verify that the user only has read permissions for SFS.
 - Choose **Service List** > **Scalable File Service**. On the SFS console, click **Create File System** in the upper right corner. If a message appears indicating that you have insufficient permissions to perform the operation, the corresponding policy is in effect.
For SFS Capacity-Oriented, the **SFS ReadOnlyAccess** policy is in effect.
For SFS Turbo, the **SFS Turbo ReadOnlyAccess** policy is in effect.
 - Choose another service from **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the corresponding policy is in effect.
For SFS Capacity-Oriented, the **SFS ReadOnlyAccess** policy is in effect.
For SFS Turbo, the **SFS Turbo ReadOnlyAccess** policy is in effect.

1.2 Creating a Custom Policy

You can create custom policies to supplement the system-defined policies of SFS. For the actions supported for custom policies, see [Permissions Policies and Supported Actions](#).

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.

For details, see [Creating a Custom 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 together with other policies. If the permissions granted to an IAM 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. Example policy denying file system deletion:

```
{
  "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"  
    ]  
  }  
]
```


2 File System Management

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.

 **NOTE**

- Viewing details of SFS Turbo file systems depends on the VPC service. Ensure that the required role or policy has been configured.
The permissions of the **SFS Turbo ReadOnlyAccess** policy already include the permissions of **VPC ReadOnlyAccess**, which are required for querying file system details. An IAM user assigned the **SFS Turbo ReadOnlyAccess** policy does not need to have the **VPC ReadOnlyAccess** policy assigned explicitly.

Procedure

Step 1 Log in to the SFS console.

Step 2 In the file system list, view the file systems you have created. [Table 2-1](#) describes the file system parameters.

Table 2-1 Parameter description

Parameter	Description
Name	Name of the file system, for example, sfs-name-001
AZ	Availability zone where the file system resides
Status	Possible values are Available, Unavailable, Frozen, Creating, Deleting .
Type	File system type
Protocol Type	File system protocol, which can be NFS or CIFS

Parameter	Description
Used Capacity (GB)	File system space already used for data storage NOTE This information is refreshed every 15 minutes. The used capacity will not be displayed if less than 1 MB of an SFS Capacity-Oriented file system is used.
Maximum Capacity (GB)	Maximum capacity of the file system
Encrypted	Encryption status of the file system. The value can be Yes or No .
Enterprise Project	Enterprise project to which the file system belongs
Mount Point	File system mount point. The format of an NFS file system is <i>File system domain name:/Path</i> or <i>File system IP address/.</i> . The format of a CIFS file system is <i>\\File system domain name\Path</i> . 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, operations include capacity expansion, deletion, monitoring metric viewing, subscription renewal, and unsubscription. For a General Purpose File System, operations include deletion only.

Step 3 (Optional) Search for the file system by file system name keyword, key ID, or file system status.

----End

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

You have unmounted the file system to be deleted. For details about how to unmount a file system, see [Unmount a File System](#).

Procedure

Step 1 Log in to the 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. Only SFS Capacity-Oriented file systems support batch deletion.

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

In the displayed dialog box for a General Purpose File System, confirm the information and click **OK**.

 **NOTE**

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

A General Purpose File System can only be deleted when the file system's used capacity and the number of files in the file system are both zero.

Figure 2-1 Deleting an SFS Turbo file system

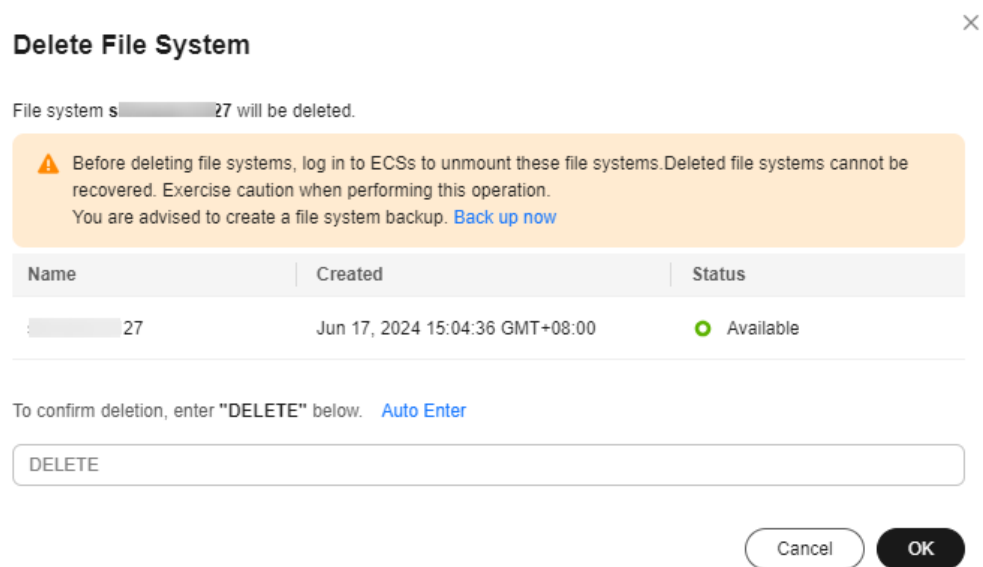
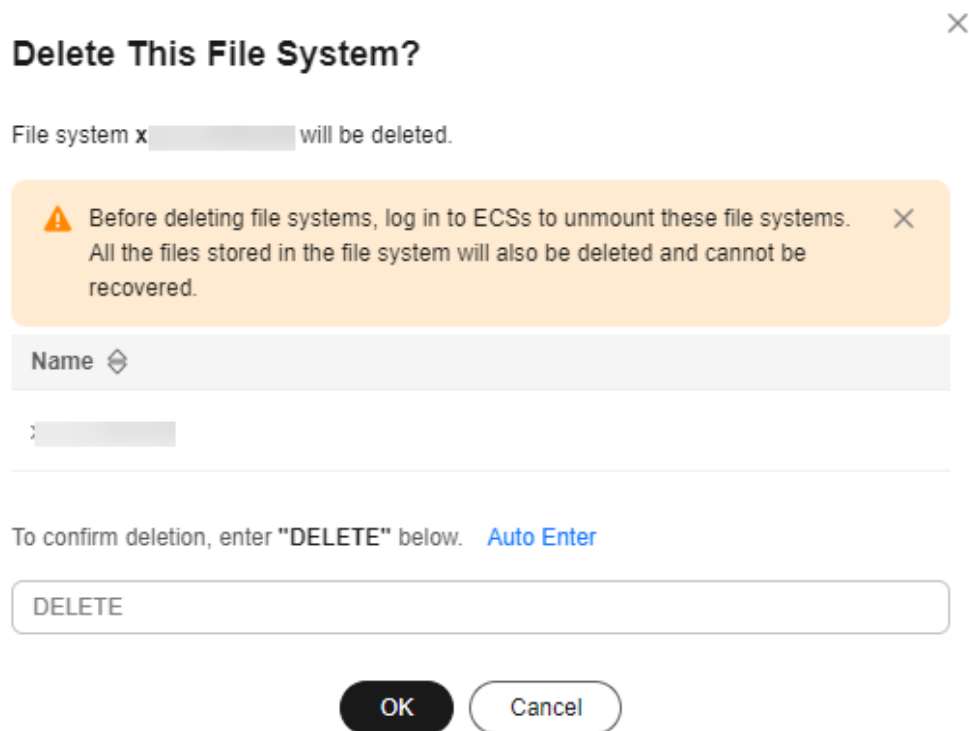


Figure 2-2 Deleting a General Purpose File System



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

----End

3 Network Configuration

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 to share files, a file system and the cloud servers need to run in the same VPC.

In addition, VPC can use network access control lists (ACLs) for access control. A network ACL is an access control policy system for one or more subnets. Based on inbound and outbound rules, the network ACL 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](#).

Scenarios

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

SFS Turbo can work with VPC Peering to allow cloud servers in two or more VPCs of the same region to share the same file system as if they are in the same VPC. For details about VPC peering connection, see [VPC Peering Connection](#).

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

Use Restrictions

- You can add a maximum of 20 authorized VPCs for a file system and a maximum of 400 ACL rules for each authorized VPC. When you add an authorized VPC, the IP address 0.0.0.0/0 will be added automatically.
- If a VPC added to a file system has been deleted from the VPC console, the IP addresses or IP address ranges 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 remove it from the list.

- Before adding an authorized VPC for a General Purpose File System, you need to create a VPC endpoint to establish communication between the compute resources and the file system.
- You need to configure a VPC endpoint for each VPC you want to add as an authorized VPC for a General Purpose File System. Or, the file system will fail to be mounted.

Procedure for SFS Capacity-Oriented

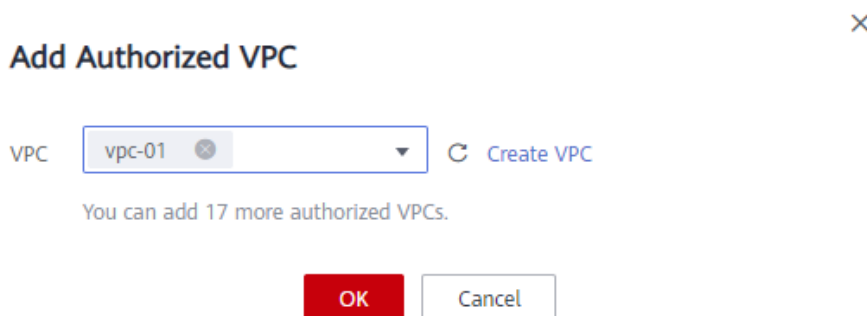
Step 1 Log in to the SFS console.

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 **Add Authorized VPC** and select a VPC on the displayed dialog box, as shown in [Figure 3-1](#). If no VPCs are available, create one and then add. You can add multiple VPCs for a file system.

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

Figure 3-1 Adding VPCs



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

Step 5 View the information about authorized VPCs in the VPC list. [Table 3-1](#) describes the parameters.

Table 3-1 Parameter description

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

Parameter	Description
Operation	Includes the Add and Deletion operations. Click Add to add an authorized address, including adding an authorized IP address, read/write permission, user permission, user root permission, and priority. For details, see Table 3-2 . Click Delete to remove this authorized VPC.


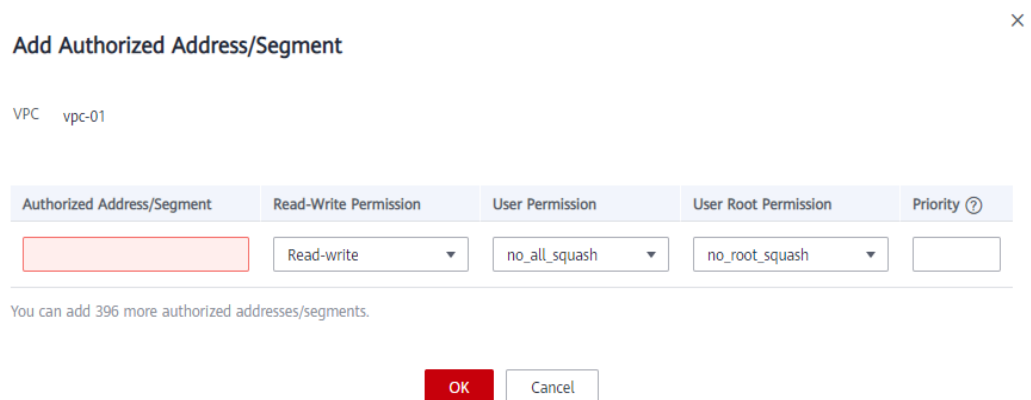
Step 6 Click  on the left of the VPC name to view the IP addresses or IP address ranges added to this VPC. You can add, edit, or delete IP addresses or IP address ranges. Click **Add** in the **Operation** column of the VPC. The **Add Authorized Address/Segment** dialog box is displayed, as shown in [Figure 3-2](#). [Table 3-2](#) describes the parameters displayed.

Figure 3-2 Adding an authorized address or segment



×

Add Authorized Address/Segment

VPC vpc-01

Authorized Address/Segment	Read-Write Permission	User Permission	User Root Permission	Priority ?
<input style="border: 2px solid red;" type="text"/>	Read-write ▼	no_all_squash ▼	no_root_squash ▼	<input type="text"/>

You can add 396 more authorized addresses/segments.

Table 3-2 Parameter description

Parameter	Description
Authorized Address/Segment	<ul style="list-style-type: none">• Enter one IPv4 address or range in each line.• Enter a valid IPv4 address or range that is not 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 to access the file system. Do not enter an IP address or IP address range starting with any number ranging from 224 to 255, for example 224.0.0.1 or 255.255.255.255, because class D and class E IP addresses are not supported. IP addresses starting with 127 are also not supported. If you enter an invalid IP address or IP address range, the authorization may fail to be added, or the added authorization does not work.• Do not enter multiple IP addresses (separated using commas) in a line. For example, do not enter 10.0.1.32,10.5.5.10.• If you enter an IP address range, enter it in the format of <i>IP address/mask</i>. For example, enter 192.168.1.0/24. Do not enter 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	You can select Read-write or Read-only . Read-write is preselected.
User Permission	Whether to retain the user identifier (UID) and group identifier (GID) of the shared directory. There are two options: <ul style="list-style-type: none">• all_squash: The UIDs and GIDs of shared files are mapped to user nobody, which is suitable for public directories.• no_all_squash (default value): The UIDs and GIDs of shared files are retained. You do not need to configure this parameter if you add an authorized address for a CIFS file system.
User Root Permission	Whether to allow the client to access as root . There are two options: <ul style="list-style-type: none">• root_squash: Clients cannot access as root. When a client accesses as root, the user is mapped to user nobody.• no_root_squash (default value): Clients are allowed to access as root who has full control and access permissions of the root directories. You do not need to configure this parameter if you add an authorized address for a CIFS file system.

Parameter	Description
Priority	<p>The value must be an integer ranging from 0 to 100. 0 has the highest priority, and 100 the lowest. In the same VPC, the permission of the IP address or IP address range with the highest priority is preferentially used. If IP addresses or IP address ranges are of the same priority, the permission of the most recently added or modified one will be used.</p> <p>For example, if the client IP address 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) is used because it has a lower priority. If there is no other priority, all IP addresses in 10.1.1.0/24, including 10.1.1.32, have the read-only permission.</p>

 **NOTE**

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

----End

General Purpose File System

- Step 1** Log in to the SFS console.
- Step 2** In the navigation pane on the left, choose **General Purpose File System** to go to its console.
- Step 3** In the file system list, click the name of the target file system to go to the file system details page.
- Step 4** In the left navigation pane, choose **Permissions Management**.
- Step 5** Click **Add Authorization Rule**. A dialog box is displayed, as shown in [Figure 3-3](#). If no VPCs are available, create one.

[Table 3-3](#) describes the parameters displayed.

Figure 3-3 Add Authorization



Table 3-3 Parameter description

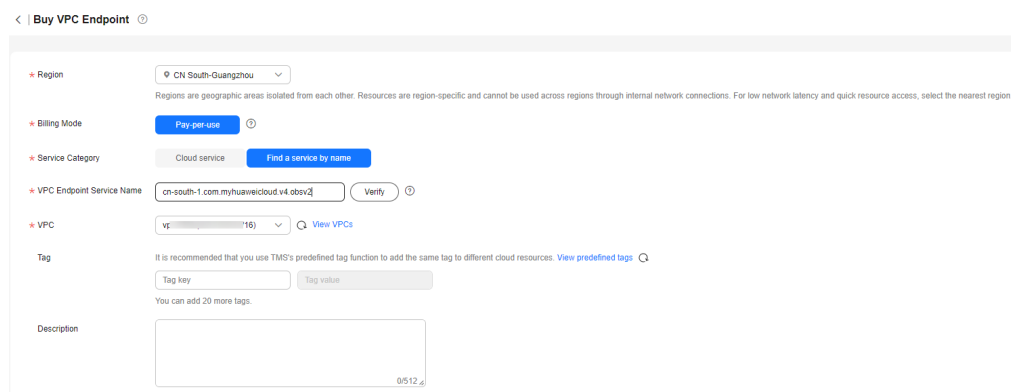
Parameter	Description
VPC	VPC you want to add, for example, vpc-30e0 . If no VPC is available, create one.
Authorizations	You can select Read/Write or Read-only . Read/Write is preselected.

Step 6 Click **OK**. The added VPC will be displayed in the list.

Step 7 On the **VPC Endpoints** page, click **Buy VPC Endpoint**.

The **Buy VPC Endpoint** page is displayed.

Figure 3-4 Buy VPC Endpoint



Step 8 Set the parameters as prompted.

Table 3-4 Parameters for purchasing an endpoint

Parameter	Description
Region	Region where the VPC endpoint is located. Ensure that this region is the same as the one where the planned General Purpose File System resides. VPC EP supports General Purpose File System only in the CN North-Beijing4, CN East-Shanghai1, and CN South-Guangzhou regions.
Billing Mode	Pay-per-use is preselected by default, but you will not be billed for the endpoint purchased for General Purpose File Systems.

Parameter	Description
Service Category	<p>Select Find a service by name.</p> <p>Enter a VPC endpoint service name based on the region selected.</p> <ul style="list-style-type: none"> If the CN North-Beijing4 region is selected, enter cn-north-4.com.myhuaweicloud.v4.storage.lz13. If the CN South-Guangzhou region (AZ1) is selected, enter cn-south-1.com.myhuaweicloud.v4.obsv2. If the CN South-Guangzhou region (AZ6) is selected, enter cn-south-1.com.myhuaweicloud.v4.obsv2.storage.lz06. If the CN East-Shanghai1 region is selected, enter cn-east-3.com.myhuaweicloud.v4.storage.lz07. <p>After entering the service name, click Verify.</p> <p>If Service name found is displayed, proceed with subsequent steps.</p> <p>If Service name not found is displayed, check whether the entered service name is correct. If the problem persists, submit a service ticket.</p>
VPC	Select the VPC you have added as authorized VPC of the General Purpose File System.
Tag	<p>Optional</p> <p>VPC endpoint tags. Each tag consists of a key and a value. You can add a maximum of 10 tags to one VPC endpoint. Tag keys and values must meet the requirements listed in Table 3-5.</p> <p>NOTE</p> <p>If a predefined tag has been created in TMS, you can select the corresponding tag key and value.</p> <p>For details about predefined tags, see Predefined Tag Overview.</p>

[Table 3-5](#) describes the tag parameters.

Table 3-5 Tag parameter description

Parameter	Description	Example Value
Tag key	<p>Each tag has a unique key. You can customize the key or select the key of an existing tag created in TMS.</p> <p>A tag key:</p> <ul style="list-style-type: none"> Can contain 1 to 36 Unicode characters. Can contain only letters, digits, hyphens (-), and underscores (_). 	Key_0001

Parameter	Description	Example Value
Tag value	A tag value can be repetitive or left blank. A tag value: <ul style="list-style-type: none">• Can contain 0 to 43 Unicode characters.• Can contain only letters, digits, hyphens (-), and underscores (_).	Value_0001

Step 9 Click **Next**.

- If you do not need to modify the specifications, click **Submit**.
- If you need to modify the specifications, click **Previous**, modify the configuration as needed, and then click **Submit**.

Step 10 Go back to the VPC endpoint list and check whether the status of the VPC endpoint changes to **Accepted**. If so, the VPC endpoint has been connected to the VPC endpoint service.

----End

Verification

After an authorized VPC is added for the file system, if the file system can be mounted to ECSs in that VPC and the ECSs can access the file system, the configuration is successful.

Example

You create an SFS Capacity-Oriented file system A in VPC-B whose CIDR block is **10.0.0.0/16**. You had an ECS D (private IP address **192.168.10.11**) in VPC-C whose CIDR block is **192.168.10.0/24**. If you want to mount file system A to ECS D and perform reads and writes in the file system from ECS D, you need to add VPC-C as an authorized VPC of file system A, add the private IP address of ECS D as an authorized address of VPC-C, and set **Read-Write Permission** to **Read-write**.

You buy a new ECS F (private IP address **192.168.10.22**) in the VPC-C whose CIDR block is **192.168.10.0/24**. If you want ECS F to have only the read permission for file system A and a lower read priority than ECS D, you need to add the private IP address of ECS F as an authorized address of VPC-C, set **Read-Write Permission** to **Read-only**, and set **Priority** to an integer ranging from 0 and 100 and greater than the priority set for ECS D.

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 the VPCs used by other accounts are added as authorized VPCs of an SFS Capacity-Oriented file system, and IP addresses or ranges of cloud servers are

added as authorized addresses, cloud servers under different accounts can share the same file system.

For more information about VPC, see the [Virtual Private Cloud](#).

With VPC peering, an SFS Turbo file system can be accessed across accounts. For details about VPC peering connection and usage instructions, see [VPC Peering Connection](#).

This section describes how to configure multi-account access for an SFS Capacity-Oriented file system. Currently, only SFS Capacity-Oriented file systems in the CN North-Beijing4 region support multi-account access.

Use Restrictions

- You can add a maximum of 20 authorized VPCs for a file system and a maximum of 400 ACL rules for each authorized VPC.
- If a VPC added to a file system has been deleted from the VPC console, the IP addresses or IP address ranges 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 remove it from the list.

Procedure for SFS Capacity-Oriented

Step 1 Log in to the 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. See [Figure 3-5](#).

Figure 3-5 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 16 more authorized VPCs.

Authorized Address/Segment

You can add 396 more authorized addresses/segments.

Priority ?

Read-Write Permission Read-write ▼

User Permission no_all_squash ▼

User Root Permission no_root_squash ▼

OK
Cancel

Table 3-6 describes the parameters to be configured.

Table 3-6 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	<ul style="list-style-type: none"> • Enter one IPv4 address or range in each line. • Enter a valid IPv4 address or range that is not 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 to access the file system. Do not enter an IP address or IP address range starting with any number ranging from 224 to 255, for example 224.0.0.1 or 255.255.255.255, because class D and class E IP addresses are not supported. IP addresses starting with 127 are also not supported. If you enter an invalid IP address or IP address range, the authorization may fail to be added, or the added authorization does not work. • Do not enter multiple IP addresses (separated using commas) in a line. For example, do not enter 10.0.1.32,10.5.5.10. • If you enter an IP address range, enter it in the format of <i>IP address/mask</i>. For example, enter 192.168.1.0/24. Do not enter 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.
Priority	<p>The value must be an integer ranging from 0 to 100. 0 has the highest priority, and 100 the lowest. In the same VPC, the permission of the IP address or IP address range with the highest priority is preferentially used. If IP addresses or IP address ranges are of the same priority, the permission of the most recently added or modified one will be used. For example, if the client IP address 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) is used because it has a lower priority. If there is no other priority, all IP addresses in 10.1.1.0/24, including 10.1.1.32, have the read-only permission.</p>
Read-Write Permission	<p>You can select Read-write or Read-only. The default value is Read-write.</p>
User Permission	<p>Whether to retain the user identifier (UID) and group identifier (GID) of the shared directory. There are two options:</p> <ul style="list-style-type: none"> • all_squash: The UIDs and GIDs of shared files are mapped to user nobody, which is suitable for public directories. • no_all_squash (default value): The UIDs and GIDs of shared files are retained. <p>You do not need to configure this parameter if you add an authorized address for a CIFS file system.</p>

Parameter	Description
User Root Permission	<p>Whether to allow the client to access as root. There are two options:</p> <ul style="list-style-type: none"> • root_squash: Clients cannot access as root. When a client accesses as root, the user is mapped to user nobody. • no_root_squash (default value): Clients are allowed to access as root who has full control and access permissions of the root directories. <p>You do not need to configure this parameter if you add an authorized address for a CIFS file system.</p>

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


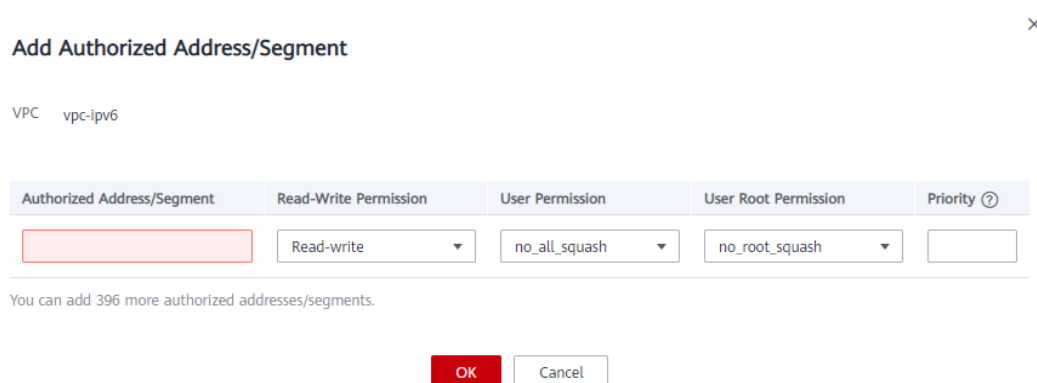
Step 5 Click  on the left of the VPC name to view the IP addresses or IP address ranges added to this VPC. You can add, edit, or delete IP addresses or IP address ranges. In the **Operation** column of the target VPC, click **Add**. The **Add Authorized Address/Segment** dialog box is displayed. See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

Figure 3-6 Adding an authorized address or segment



----End

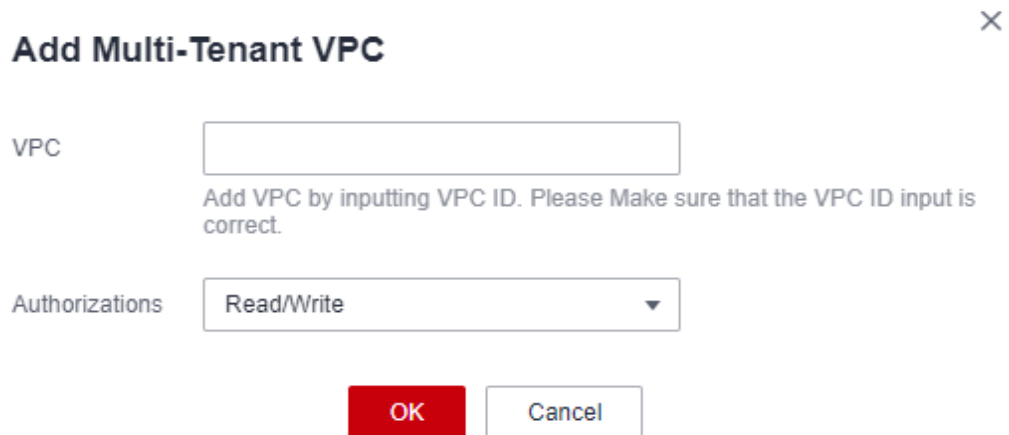
Procedure for General Purpose File System

Step 1 Log in to the SFS console.

Step 2 Go to the General Purpose File System list, locate the target file system, and click the file system name. On the displayed details page, go to the permissions management tab page.

Step 3 Click **Add Multi-Tenant VPC** to add VPCs of other accounts. See [Figure 3-7](#).

Figure 3-7 Add Multi-Tenant VPC



Add Multi-Tenant VPC ×

VPC

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

Authorizations

Step 4 Click **OK**. The added VPCs are displayed in the list.

----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 Configuring DNS

A DNS server is used to resolve domain names of file systems. For details about DNS server IP addresses, see [What Are Private DNS Servers and What Are Their Addresses?](#)

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.

Windows Server 2012 is used as an example in the operation procedures for Windows.

Procedure (Linux)

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 100.125.1.250
nameserver 104.104.104.104
nameserver 104.104.105.105
```

The format is as follows:

```
nameserver 100.125.1.250
```

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@cloud11174-File-Test ~]# lsattr /etc/resolv.conf
----i-----e- /etc/resolv.conf
```

----End

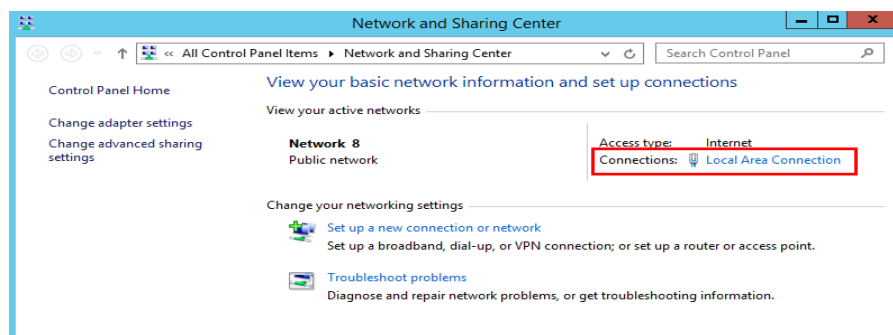
Procedure (Windows)

Step 1 Go to the ECS console and log in to the ECS running Windows Server 2012.

Step 2 Click **This PC** in the lower left corner.

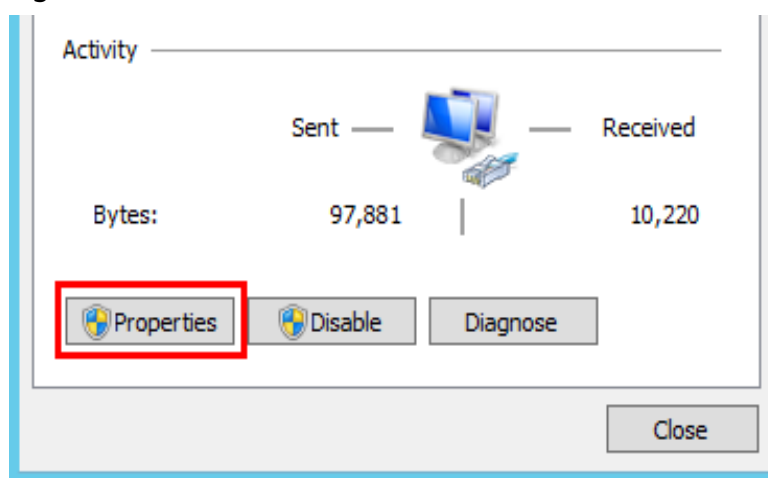
Step 3 On the page that is displayed, right-click **Network** and choose **Properties** from the drop-down list. The **Network and Sharing Center** page is displayed, as shown in **Figure 3-10**. Click **Local Area Connection**.

Figure 3-10 Page for network and sharing center



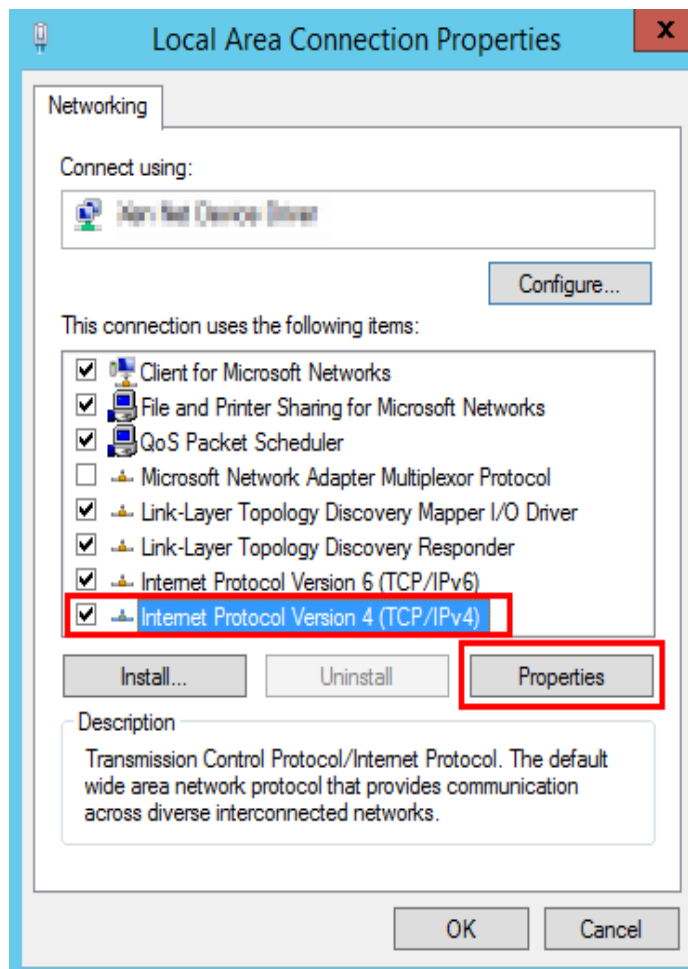
Step 4 In the **Activity** area, select **Properties**. See [Figure 3-11](#).

Figure 3-11 Local area connection



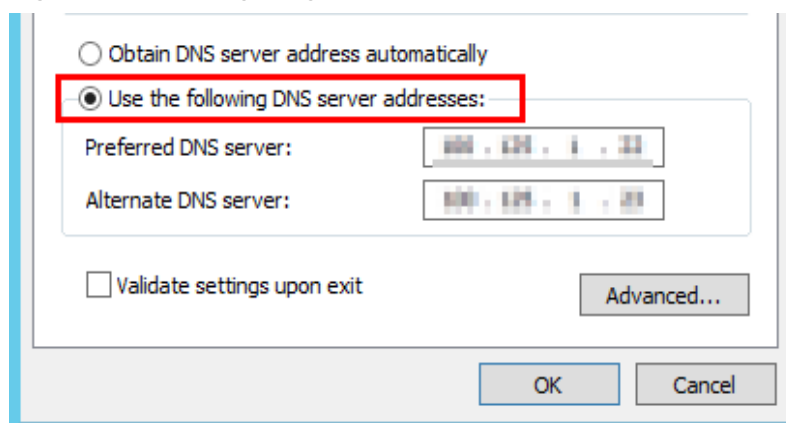
Step 5 In the **Local Area Connection Properties** dialog box that is displayed, select **Internet Protocol Version 4 (TCP/IPv4)** and click **Properties**. See [Figure 3-12](#).

Figure 3-12 Local area connection properties



Step 6 In the dialog box that is displayed, select **Use the following DNS server addresses:** and configure DNS, as shown in [Figure 3-13](#). The DNS server IP address is 100.125.1.250. After completing the configuration, click **OK**.

Figure 3-13 Configuring DNS on Windows



----End

4 File System Resizing

Scenarios

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

Notes and Constraints

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

SFS Turbo file systems support online capacity expansion, during which mounting a file system may fail and the connection being used for mounting will experience about a 30-second (max. 3 minutes) I/O delay. So you are advised to expand capacity during off-peak hours. Note that only **In-use** file systems can be expanded.

The capacity of an SFS Turbo file system cannot be decreased. You can purchase a new file system with a smaller capacity and migrate your data to the new file system.

General Purpose File Systems have no capacity limit and do not support resizing.

Precautions

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 \leq (Capacity quota of the cloud account - Total capacity of all the other file systems owned by the cloud account)
 - For example, a cloud account 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 \geq Used capacity of the file system

For example, a cloud account 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 a new capacity smaller than 10 TB.

Expanding Capacity of a Yearly/Monthly SFS Turbo File System

Step 1 Log in to the management console and choose **Storage > Scalable File Service**.

Step 2 In the file system list, locate the SFS Turbo file system you want to expand capacity and click **Expand Capacity** in the **Operation** column to go to the **Expand Capacity** page.

Figure 4-1 Expanding capacity of a yearly/monthly SFS Turbo file system

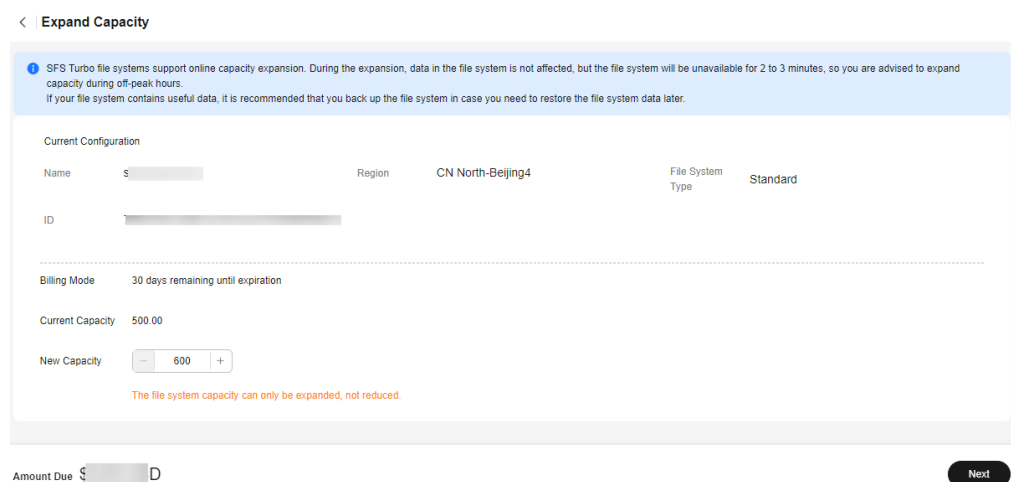


Table 4-1 Capacity expansion parameters

Parameter	Description
Current Capacity	Current storage capacity of the file system
New Capacity	<p>New storage capacity of the file system</p> <p>Constraints:</p> <ul style="list-style-type: none"> For a Standard, Standard – Enhanced, Performance, or Performance – Enhanced file system, the expansion increment is 100 GB. A Standard or Performance file system can be expanded to up to 32 TB, and a Standard - Enhanced or Performance - Enhanced file system can be expanded to up to 320 TB.

Step 3 Enter the new capacity based on service requirements and then click **Next**.

Step 4 Confirm the resource information and click **Submit**.

Step 5 Complete the payment as instructed and return to the file system list. Click the name of the expanded file system and check that the capacity has been expanded.

----End

Expanding Capacity of a Pay-per-Use SFS Turbo File System

Step 1 Log in to the management console and choose **Storage > Scalable File Service**.

Step 2 In the file system list, locate the SFS Turbo file system you want to expand capacity and click **Expand Capacity** in the **Operation** column to go to the **Expand Capacity** page.

Figure 4-2 Expanding capacity of a pay-per-use SFS Turbo file system

✕

Expand Capacity

i SFS Turbo file systems support online capacity expansion. During the expansion, data in the file system is not affected, but the file system will be unavailable for 2 to 3 minutes, so you are advised to expand capacity during off-peak hours.
If your file system contains useful data, it is recommended that you back up the file system in case you need to restore the file system data later.

Current Configuration

Name	ε [redacted]
ID	64 [redacted]
Region	CN North-Beijing4
AZ	AZ1
File System Type	: MB/s/TiB

Billing Mode	Pay-per-use (Price: {0}/hour)
Current Capacity	[redacted] 3B
New Capacity	<input type="text" value=""/> TB

The file system capacity can only be expanded, not reduced.

Price: \$[redacted]/hour ? OK

Step 3 Enter the new capacity based on service requirements. For detailed parameter descriptions, see [Table 4-1](#).

Step 4 Click **OK**. In the file system list, check that the file system capacity has been expanded.

----End

5 Quotas

What Is Quota?

A quota is a limit on the quantity or capacity of a certain type of service resources that you can use, for example, the maximum number of SFS file systems that you can create.

If a quota cannot meet your needs, apply for a higher quota.

How Do I View My Quotas?


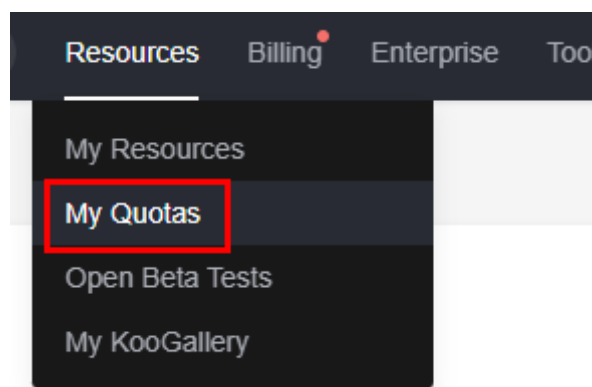
1. Log in to the management console.
2. Click  in the upper left corner and select your desired region and project.
3. In the upper right corner of the page, choose **Resources > My Quotas**.
The **Service Quota** page is displayed.

Figure 5-1 My Quotas



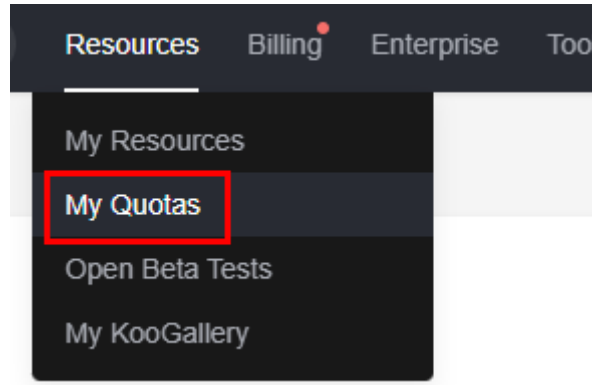
4. View the used and total quota of each type of resources on the displayed page.
If a quota cannot meet service requirements, apply for a higher quota.

How Do I Apply for a Higher Quota?

1. Log in to the management console.

- In the upper right corner of the page, choose **Resources > My Quotas**.
The **Service Quota** page is displayed.

Figure 5-2 My Quotas



- Click **Increase Quota** in the upper right corner of the page.

Figure 5-3 Increasing quota

Service	Resource Type	Used Quota	Total Quota
Auto Scaling	AS group	0	
	AS configuration	0	
Image Management Service	Image	0	
Cloud Container Engine	Cluster	0	
FunctionGraph	Function	0	
	Code storage(MB)	0	
Elastic Volume Service	Disk	3	
	Disk capacity(OB)	120	
	Snapshots	4	
Storage Disaster Recovery Service	Protection group	0	
	Replication pair	0	
Cloud Server Backup Service	Backup Capacity(OB)	0	
	Backup	0	
Scalable File Service	File system	0	
	File system capacity(OB)	0	
CDN	Domain name	0	
	File URL refreshing	0	
	Directory URL refreshing	0	
	URL prewarming	0	

- On the **Create Service Ticket** page, configure parameters as required.
In the **Problem Description** area, fill in the content and reason for adjustment.
- After all necessary parameters are configured, select **I have read and agree to the Ticket Service Protocol and Privacy Statement** and click **Submit**.

6 Encryption

Creating an Encrypted File System

To use the file system encryption function, you need to authorize SFS Capacity-Oriented to access KMS when creating an SFS Capacity-Oriented file system. If you have the Security Administrator rights, grant SFS the permissions to access KMS directly. Otherwise, you need to contact the system administrator to obtain the "Security Administrator" rights first. For details, see [File System Encryption](#).

For SFS Turbo file systems, no authorization is required.

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](#).

7 Backup

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

The new backup function is now available. The original backup data will be automatically cleared. Go to the CBR page and back up your data there in a timely manner to avoid data loss.

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.

If a file system is faulty or encounters a logical error (for example, accidental 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

When a previous-generation SFS Turbo file system (Standard, Standard - Enhanced, Performance, or Performance - Enhanced) is being backed up, mounting the file system may fail. This is because the connection used for mounting may experience an I/O delay about 30 seconds. You are advised to perform backup during off-peak hours.

- Step 1** Log in to the CBR console.
- Step 2** In the navigation pane on the left, choose **SFS Turbo Backups**.
- Step 3** Buy a backup vault by following the instructions in [Purchasing an SFS Turbo Backup Vault](#). Then, create a backup by following the instructions in [Creating an SFS Turbo Backup](#).
- Step 4** Wait for CBR to automatically create a file system backup.

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 Create a new file system from the backup if the file system becomes faulty or encounters an error occurred. 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 SFS Turbo file system. Data on the new file system is the same as that in the backup.

NOTE

You can only create pay-per-use SFS Turbo file systems from backups. To create yearly/monthly ones from backups, you need to first create the pay-per-use file systems and then change their billing modes to yearly/monthly.

Step 1 Log in to the 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 Click **Create File System** in the **Operation** column of the backup. The button is available only when the backup status is **Available**.

NOTE

For how to create backups, see [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](#).
- You can change the storage class of the file system within a certain range. For example, you can change a file system from Standard to Performance, but cannot from Standard to Standard - Enhanced.
- The billing mode of the new file system can only be pay-per-use.

Step 5 Click **Next**.

Step 6 Confirm the file system information and click **Submit**.

Step 7 Pay the fees as prompted and click **OK**.

Step 8 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

8 Monitoring

8.1 SFS Metrics

Function

This section describes metrics reported by 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 8-1 SFS Capacity-Oriented (sold-out) 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	File system	4 minutes
write_bandwidth	Write Bandwidth	Write bandwidth of a file system within a monitoring period Unit: byte/s	≥ 0 bytes/s	File system	4 minutes

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

Table 8-2 General Purpose File System 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	File system	4 minutes
write_bandwidth	Write Bandwidth	Write bandwidth of a file system within a monitoring period Unit: byte/s	≥ 0 bytes/s	File system	4 minutes
read_tps	Read TPS	Number of read operations of a file system within a monitoring period Unit: count/s	≥ 0 counts/s	File system	4 minutes

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
write_tps	Write TPS	Number of write operations of a file system within a monitoring period Unit: count/s	≥ 0 counts/s	File system	4 minutes

 **NOTE**

The used capacity of a General Purpose File System is currently not monitored.

Dimension

Key	Value
share_id	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 & Governance > Cloud Eye > Cloud Service Monitoring > Scalable File Service**. In the file system list, locate the target file system and click **View Metric** in the **Operation** column.

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

Figure 8-1 shows the monitoring graphs. For more information about Cloud Eye, see the *Cloud Eye User Guide*.

Figure 8-1 Monitoring graphs of SFS Capacity-Oriented



----End

8.2 SFS Turbo Metrics

Function

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

Namespace

SYS.EFS

Metrics

Table 8-3 SFS Turbo metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
client_connections	Client Connections	Number of client connections NOTE Only active client connections are counted. A network connection is automatically disconnected when the client has no I/Os for a long time and is automatically re-established when there are I/Os.	≥ 0	SFS Turbo file system	1 minute
data_read_io_bytes	Read Bandwidth	Data read I/O load Unit: byte/s	≥ 0 bytes/s	SFS Turbo file system	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
data_write_io_bytes	Write Bandwidth	Data write I/O load Unit: byte/s	≥ 0 bytes/s	SFS Turbo file system	1 minute
metadata_io_bytes	Metadata Read and Write Bandwidth	Metadata read and write I/O load Unit: byte/s	≥ 0 bytes/s	SFS Turbo file system	1 minute
total_io_bytes	Total Bandwidth	Total I/O load Unit: byte/s	≥ 0 bytes/s	SFS Turbo file system	1 minute
iops	IOPS	I/O operations per unit time	≥ 0	SFS Turbo file system	1 minute
used_capacity	Used Capacity	Used capacity of a file system Unit: byte	≥ 0 bytes	SFS Turbo file system	1 minute
used_capacity_percent	Capacity Usage	Percentage of used capacity in the total capacity Unit: percent	0% to 100%	SFS Turbo file system	1 minute
used_inode	Used inodes	Number of inodes used in a file system	≥ 1	SFS Turbo file system	1 minute
used_inode_percent	Inode Usage	Percentage of used inodes to total inodes in a file system Unit: percent	0% to 100%	SFS Turbo file system	1 minute

Dimension

Key	Value
efs_instance_id	Instance

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 & Governance > Cloud Eye > Cloud Service Monitoring > SFS Turbo**. In the file system list, click **View Metric** in the **Operation** column of the target file system.

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

Figure 8-2 shows the monitoring graphs. For more information about Cloud Eye, see the *Cloud Eye User Guide*.

Figure 8-2 SFS Turbo monitoring graphs



----End

8.3 Creating Alarm Rules

The alarm function is based on collected metrics. You can set alarm rules for key metrics of SFS. When the metric data triggers the conditions set in the alarm rule, Cloud Eye sends emails to you, or sends HTTP/HTTPS requests to the servers. In this way, you are immediately informed of cloud service exceptions and can quickly handle the faults to avoid service losses.

Cloud Eye uses Simple Message Notification (SMN) to notify users. This requires you to create a topic and add relevant subscribers for this topic on the SMN console first. Then when you create alarm rules, you can enable the **Alarm Notification** function and select the created topic. When an error occurs, Cloud Eye can broadcast alarm information to those subscribers in real time.

Creating an Alarm Rule

1. Log in to the management console.
2. Choose **Management & Governance > Cloud Eye > Cloud Service Monitoring > Scalable File Service**. Or, choose **Management & Governance > Cloud Eye > Cloud Service Monitoring > Elastic File Service**.
3. Click **Create Alarm Rule** in the **Operation** column of the target file system.
4. On the **Create Alarm Rule** page, set parameters as prompted.
 - a. Select an object and configure other parameters listed in **Table 8-4**. Click **Next**.

 NOTE

If the monitored object is a file system, you can only search it by ID instead of name.

Table 8-4 Parameter description

Parameter	Description	Example Value
Resource Type	Specifies the name of the service for which the alarm rule is configured.	Scalable File Service
Dimension	Specifies the metric dimension of the alarm rule.	File systems
Monitored Object	Specifies the resource for which the alarm rule is configured. You can specify one or more resources.	-

- b. In the **Select Metric** step, select **Import from template** and configure parameters based on [Table 8-5](#).

Table 8-5 Parameter description

Parameter	Description	Example Value
Source	Specifies the means by which you create the alarm rule.	Import from template
Template	Select the template to be imported.	-
Send Notification	Specifies whether to notify users when alarms are triggered. Notifications can be sent via emails, or HTTP/HTTPS requests. You can enable (recommended) or disable this function.	Enable
Notification Object	Name of the topic to which the alarm notification is sent. If you enable the notification function, you need to select a topic. If no desired topics are available, you need to create one first, whereupon the SMN service is invoked. For details about how to create a topic, see the <i>Simple Message Notification User Guide</i> .	-

Parameter	Description	Example Value
Trigger Condition	Specifies the condition for triggering the alarm. You can select Generated alarm , Cleared alarm , or both.	-

- c. In the **Specify Rule Name** step, set the parameters listed in [Table 8-6](#). After the configuration is complete, click **Create**.

Table 8-6 Parameter description

Parameter	Description	Example Value
Name	Name of the alarm rule. The system generates a name randomly but you can change it.	alarm-b6al
Description	Alarm rule description. This parameter is optional.	-

After the alarm rule is created, if the metric data reaches the specified threshold, Cloud Eye immediately informs you that an exception has occurred. For details about other operations, see the *Cloud Eye User Guide*.

9 Auditing

9.1 Supported SFS Operations

Scenarios

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

Currently, only SFS Turbo file systems support recording of resource operations using CTS. SFS Capacity-Oriented file systems do not support this function.

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 Getting Started*.

Operations

Table 9-1 SFS operations traced by CTS

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

Operation	Resource Type	Trace
Expanding a shared file system	sfs	extendShare
Shrinking a shared file system	sfs	shrinkShare

Table 9-2 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 & Governance > 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 Click  in the upper left corner and select a region and project.

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

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

Step 4 Click **Trackers** in the left pane.

Step 5 Click **Disable** on the right of the tracker information.

Step 6 Click **Yes**.

Step 7 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**

10 Typical Applications

10.1 High-performance Computing

Context

A high-performance computing (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 is widely used. When the software is operating, compute nodes need to communicate with each other closely, which requires a file system that can provide high bandwidth and low latency.
- **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 the file system to provide large memory and high bandwidth.
- **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 to 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. 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 cost and O&M cost are greatly reduced, the service rollout time is shortened, and efficiency is boosted.

Configuration Process

1. Prepare the files of DNA sequencing to be uploaded.
2. Log in to the SFS console. Create a file system to store the files of DNA sequencing.
3. Log in to the cloud 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.
- Cloud servers that function as head nodes and compute nodes have been created, and are in the created VPC. For details about how to upload on-premises gene sequencing files to SFS Capacity-Oriented, see [Migrating Data Using Direct Connect](#).
- SFS has been enabled.

Example Configuration

Step 1 Log in to the SFS console.

Step 2 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

10.2 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. For details about how to upload on-premises material files to SFS Capacity-Oriented, see [Migrating Data Using Direct Connect](#).
- SFS has been enabled.

Example Configuration

Step 1 Log in to the SFS console.

Step 2 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

10.3 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 the 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. For details about how to upload on-premises website files to SFS Turbo, see [Migrating Data Using Direct Connect](#).

- SFS has been enabled.

Example Configuration

Step 1 Log in to the SFS console.

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

10.4 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 the 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. For details about how to upload on-premises log files to SFS Turbo, see [Migrating Data Using Direct Connect](#).

- SFS has been enabled.

Example Configuration

Step 1 Log in to the SFS console.

Step 2 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

11 Other Operations

11.1 Testing SFS Turbo Performance

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

Prerequisites

fiio has been installed on the cloud server. 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 fiio

The following uses a Linux CentOS system as an example:

1. Download fiio.
yum install fiio
2. Install the libaio engine.
yum install libaio-devel
3. Check the fiio version.
fiio --version

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 cloud servers 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

- fio command:

```
fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=/mnt/nfs/test_fio --bs=1M --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 11:48:57 2020
read: IOPS=7423, BW=28.0MiB/s (30.4MB/s)(7167MiB/247160msc)
slat (msec): min=1234, max=397477, avg=3145.45, stdev=3344.48
clat (msec): min=245, max=133325, avg=11162.10, stdev=12136.31
lat (msec): min=252, max=133330, avg=11166.32, stdev=12136.34
clat percentiles (msec):
| 1.00th=[ 2245],  5.00th=[ 2540], 10.00th=[ 2671], 20.00th=[ 2900],
| 30.00th=[ 3130], 40.00th=[ 3450], 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=[79160], 99.95th=[85459],
| 99.99th=[90042]
bw ( KIB/s): min=16600, max=45560, per=100.00%, avg=29696.00, stdev=5544.46, samples=494
iops   : min= 4150, max=11390, avg=7424.01, stdev=1306.11, samples=494
write: IOPS=3182, BW=12.4MiB/s (13.0MB/s)(3073MiB/247160msc)
slat (msec): min=1400, max=302730, avg=4613.59, stdev=3359.60
clat (msec): min=1447, max=140666, avg=14166.05, stdev=13373.72
lat (msec): min=1457, max=140671, avg=14170.73, stdev=13373.74
clat percentiles (msec):
| 1.00th=[  41],  5.00th=[  41], 10.00th=[  41], 20.00th=[  51],
| 30.00th=[  51], 40.00th=[  61], 50.00th=[  81], 60.00th=[ 141],
| 70.00th=[ 101], 80.00th=[ 241], 90.00th=[ 331], 95.00th=[ 421],
| 99.00th=[ 591], 99.50th=[ 671], 99.90th=[ 871], 99.95th=[ 941],
| 99.99th=[ 1221]
bw ( KIB/s): min= 7144, max=19600, per=100.00%, avg=12730.90, stdev=2395.77, samples=744
iops   : min= 1706, max= 4900, avg=3182.70, stdev=590.96, samples=744
lat (msec) : 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=1034036,706004,0,0 short=0,0,0,0 dropped=0,0,0,0
latency   : target=0, window=0, percentile=100.00%, depth=120

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=247160-247160msc
  WRITE: bw=12.4MiB/s (13.0MB/s), 12.4MiB/s-12.4MiB/s (13.0MB/s-13.0MB/s), io=3073MiB (3222MB), run=247160-247160msc
```

- fio command:

```
fio --randrepeat=1 --ioengine=libaio --name=test -output=output.log --direct=1 --filename=/mnt/nfs/test_fio --bs=1M --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=20350: Mon Jun 8 11:57:14 2020
read: IOPS=5065, BW=19.8MiB/s (20.7MB/s)(3073MiB/155200msec)
slat (nsec): min=1271, max=269500, avg=4073.51, stdev=3040.12
clat (usec): min=226, max=80185, avg=5711.35, stdev=7079.46
lat (usec): min=232, max=80187, avg=5715.49, stdev=7079.48
clat percentiles (usec):
| 1.00th=[ 1221], 5.00th=[ 1950], 10.00th=[ 2100], 20.00th=[ 2442],
| 30.00th=[ 2606], 40.00th=[ 2802], 50.00th=[ 2999], 60.00th=[ 3220],
| 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=[66047]
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=720.27, samples=310
write: IOPS=11.8k, BW=46.2MiB/s (48.4MB/s)(7167MiB/155200msec)
slat (nsec): min=1396, max=390604, avg=4405.68, stdev=3091.75
clat (usec): min=857, max=140259, avg=8377.47, stdev=8400.15
lat (usec): min=867, max=140264, avg=8382.02, stdev=8400.16
clat percentiles (nsec):
| 1.00th=[  31], 5.00th=[  41], 10.00th=[  41], 20.00th=[  41],
| 30.00th=[  51], 40.00th=[  51], 50.00th=[  51], 60.00th=[  61],
| 70.00th=[  71], 80.00th=[ 131], 90.00th=[ 211], 95.00th=[ 201],
| 99.00th=[ 421], 99.50th=[ 471], 99.90th=[ 601], 99.95th=[ 601],
| 99.99th=[ 1201]
bw ( KIB/s): min=32224, max=67456, per=99.90%, avg=47254.23, stdev=6792.41, samples=310
iops      : min= 8056, max=16864, avg=11813.55, stdev=1690.11, 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=0.61%
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 ruts: total=706597,1834043,0,0 short=0,0,0,0 dropped=0,0,0,0
latency  : target=0, window=0, percentile=100.00%, depth=120

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=155200-155200msec
WRITE: bw=46.2MiB/s (48.4MB/s), 46.2MiB/s-46.2MiB/s (48.4MB/s-48.4MB/s), io=7167MiB (7516MB), run=155200-155200msec
```

Sequential read IOPS

- fio command:

```
fio --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --
group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --rw=read
--bs=4k --size=1G --iodepth=128 --runtime=120 --numjobs=10
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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=13950.09
lat (usec): min=371, max=131118, avg=13257.29, stdev=13950.09
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=30615.41, stdev=7703.32, samples=543
iops      : min= 4664, max=15394, avg=9653.02, stdev=1925.03, 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=913130, 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 ruts: total=2621440,0,0,0 short=0,0,0,0 dropped=0,0,0,0
latency  : target=0, window=0, percentile=100.00%, depth=120

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=271519-271519msec
```

Random read IOPS

- fio command:

```
fio --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --
group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --
```

```
rw=randread --bs=4k --size=1G --iodepth=128 --runtime=120 --
numjobs=10
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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, agrb=18597KB/s, minb=18597KB/s, maxb=18597KB/s, mint=563832msec, maxt=563832msec
```

Sequential write IOPS

- fio command:

```
fio --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --
group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --
rw=write --bs=4k --size=1G --iodepth=128 --runtime=120 --numjobs=10
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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)(10.06GiB/237436msec)
slat (msec): min=1483, max=368726, avg=4388.87, stdev=3688.87
clat (usec): min=1953, max=186548, avg=11588.61, stdev=5876.84
lat (usec): min=1959, max=186552, avg=11593.86, 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=[49821], 99.95th=[53748],
| 99.99th=[69731]
bw ( KiB/s): min=31712, max=52431, per=99.99%, avg=44158.84, stdev=3987.31, samples=474
iops       : min= 7928, max=13187, avg=11839.58, stdev=996.83, samples=474
lat (msec) : 2=0.01%, 4=1.00%, 10=51.94%, 20=38.58%, 50=0.39%
lat (msec) : 100=0.00%, 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=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):
WRITE: bw=43.1MiB/s (45.2MB/s), 43.1MiB/s-43.1MiB/s (45.2MB/s-45.2MB/s), io=10.06GiB (10.7GB), run=
```

Random write IOPS

- fio command:
fio --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --rw=randwrite --bs=4k --size=1G --iodepth=128 --runtime=120 --numjobs=10

 NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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=28962: Mon Jun 8 14:37:48 2020
read: IOPS=398, BW=391MiB/s (489MB/s)(10.0GiB/26221msec)
slat (usec): min=78, max=595, avg=99.58, stdev=39.89
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=[ 188],
| 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=[ 586], 99.90th=[ 535], 99.95th=[ 542],
| 99.99th=[ 542]
bw ( KiB/s): min=381856, max=768000, per=99.52%, avg=397987.65, stdev=81583.56, samples=52
iops : min= 294, max= 758, 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=0.80%, sys=4.21%, ctx=18395, 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 (489MB/s), 391MiB/s-391MiB/s (489MB/s-489MB/s), io=10.0GiB (10.7GB), run=26221-26221msec
```

Random read bandwidth

- fio command:

```
fio --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --rw=randread --bs=1M --size=10G --iodepth=128 --runtime=120 --numjobs=1
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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 --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --rw=write --bs=1M --size=10G --iodepth=128 --runtime=120 --numjobs=1
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- fio result:

```
test: (groupid=0, jobs=1): err= 0: pid=21889: 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=[ 760], 90.00th=[ 902], 95.00th=[ 970],
| 99.00th=[ 1036], 99.50th=[ 1045], 99.90th=[ 1070], 99.95th=[ 1099],
| 99.99th=[ 1099]
bw ( KiB/s): min= 4896, max=468992, per=100.00%, avg=249580.99, 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 --ioengine=libaio --direct=1 --fallocate=none --time_based=1 --
group_reporting=1 --name=iops_fio --directory=/mnt/sfs-turbo/ --
rw=randwrite --bs=1M --size=10G --iodepth=128 --runtime=120 --
numjobs=1
```

NOTE

Variable `/mnt/sfs-turbo/` is the local path where the file to be tested is stored. Set it to the actual file name.

- 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=156000KB/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=[ 97],
| 30.00th=[ 97], 40.00th=[ 98], 50.00th=[ 98], 60.00th=[ 99],
| 70.00th=[ 99], 80.00th=[ 100], 90.00th=[ 100], 95.00th=[ 101],
| 99.00th=[ 102], 99.50th=[ 102], 99.90th=[ 103], 99.95th=[ 104],
| 99.99th=[ 195]
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
```

11.2 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 to mount a file system. However, if the permissions of user **root** are assigned to

other users, such users can also run the **mount** command to mount the file system. The following describes how to mount a file system to a Linux ECS as a common user. EulerOS is used as in this 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.
- The mount point of the file system has been obtained.

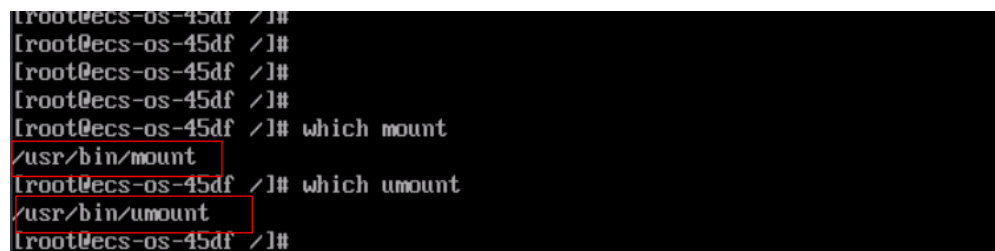
Procedure

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 11-1 Viewing command paths



```
[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 this example, user **Mike** is added.

Figure 11-2 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 11-1](#).

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

Table 11-1 Parameter description

Parameter	Description
<i>Mount point</i>	The format of an SFS Capacity-Oriented file system is <i>File system domain name:/Path</i> , for example, example.com:/share-xxx . The format of an SFS Turbo file system is <i>File system IP address/</i> , 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.
<i>Local path</i>	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

11.3 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 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, for example, */local_path*. Specify the local path used for mounting the root directory.

Step 2 Mount the subdirectory to the ECSs that are in the same VPC as the file system. You can now mount the file system to Linux ECSs using NFSv3 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*
 - General Purpose File System: *example.com:/share-xxx/subdirectory*
 - SFS Turbo: *xx.xx.xx.xx:/subdirectory*
- *Subdirectory*: Specify the subdirectory created in the previous step.
- Variable *Local path* is an ECS local directory where the file system will be mounted, for example, */local_path*. Specify the local path used for mounting the root directory.

Step 3 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 mount is successful, check whether you can access the subdirectory on the ECSs to read or write data.

----End

Troubleshooting

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

Figure 11-3 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 11-4 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]#
```

11.4 Data Migration

11.4.1 Migration Description

By default, an SFS Turbo file system can only be accessed by ECSs or CCE containers 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.
- Access from a different VPC under the same account and in the same region: Use VPC peering.

- Access from a different account in the same region: Use VPC peering.
- Access from a different region: Use Cloud Connect.

Data can be migrated to SFS Turbo by 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 Huawei Cloud ECS via the Internet.

Using the Internet to Migrate Data

11.4.2 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](#).

11.4.3 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

11.4.4 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.

Notes 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.

Resource Planning

Table 11-2 describes the resource planning in this solution.

Table 11-2 Resource planning

Resource	Example Configuration	Description
ECS	Specifications: 8 vCPUs 16 GB c7.2xlarge.2 OS: Linux Region: CN-Hong Kong VPC: VPC1	Ensure that the <code>/mnt/src</code> and <code>/mnt/dst</code> directories have been created.

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 Download and install rclone. For the download address, see <https://rclone.org/downloads/>.

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:

- **/mnt/src**: source path
- **/mnt/dst**: destination path
- **--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