

Scalable File Service

API Reference

Issue 01
Date 2024-12-09



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 Before You Start.....	1
1.1 Overview.....	1
1.2 API Calling.....	1
1.3 Endpoints.....	1
1.4 Constraints.....	1
1.5 Concepts.....	2
2 API Overview.....	4
3 Calling APIs.....	6
3.1 Making an API Request.....	6
3.2 Authentication.....	10
3.3 Response.....	12
4 Calling General Purpose File System APIs.....	14
4.1 Constructing a Request.....	14
4.2 Authentication.....	17
4.2.1 User Signature Authentication.....	17
4.2.2 Authentication of Signature in a Header.....	19
4.2.3 Signature Generators.....	26
4.3 Returned Values.....	27
5 Getting Started (SFS Capacity-Oriented).....	29
6 Getting Started with SFS Turbo.....	31
7 Getting Started with General Purpose File System.....	33
7.1 Creating a File System.....	33
7.2 Listing File Systems.....	36
8 SFS Capacity-Oriented APIs.....	39
8.1 API Version Queries.....	39
8.1.1 Querying All API Versions.....	39
8.1.2 Querying Details About an API Version.....	42
8.2 File Systems.....	46
8.2.1 Creating a Shared File System.....	46
8.2.2 Querying All Shared File Systems.....	54

8.2.3 Querying Details About All Shared File Systems.....	58
8.2.4 Querying Details About a Shared File System.....	64
8.2.5 Querying Mount Locations of a Shared File System.....	69
8.2.6 Modifying a Shared File System.....	71
8.2.7 Deleting a Shared File System.....	77
8.3 File System Access Rules.....	78
8.3.1 Adding a File System Access Rule.....	78
8.3.2 Deleting a File System Access Rule.....	82
8.3.3 Querying File System Access Rules.....	84
8.4 Quota Management.....	87
8.5 Expansion and Shrinking.....	90
8.5.1 Expanding a Shared File System.....	90
8.5.2 Shrinking a Shared File System.....	92
8.6 Tag Management.....	94
8.6.1 Adding a Tag to a Shared File System.....	94
8.6.2 Deleting a Tag from a Shared File System.....	96
8.6.3 Querying Tags of a Shared File System.....	98
8.6.4 Querying Tags of All File Systems of a Tenant.....	100
8.6.5 Batch Adding Tags to a Shared File System.....	101
8.6.6 Batch Deleting Tags from a Shared File System.....	104
8.6.7 Querying Shared File Systems by Tag.....	106
8.6.8 Querying the Number of Shared File Systems by Tag.....	113
8.7 AZ.....	119
8.7.1 Querying Availability Zones.....	120
9 SFS Turbo APIs.....	123
9.1 Lifecycle Management.....	123
9.1.1 Creating a File System.....	123
9.1.2 Querying Details About a File System.....	132
9.1.3 Deleting a File System.....	137
9.1.4 Obtaining the File System List.....	138
9.1.5 Expanding the Capacity of a File System.....	144
9.2 Connection Management.....	147
9.2.1 Changing the Security Group Associated with a File System.....	147
9.3 Tag Management.....	149
9.3.1 Adding a Tag for a File System.....	149
9.3.2 Querying Tags of a File System.....	151
9.3.3 Deleting a Tag of a File System.....	153
9.3.4 Batch Adding Tags to a File System.....	155
9.3.5 Querying Tags of All File Systems of a Tenant.....	158
9.4 Name Management.....	159
9.4.1 Changing the Name of a File System.....	160
9.5 File System Management.....	161

9.5.1 Creating an Asynchronous Task for a File System.....	161
9.5.2 Obtaining Asynchronous Tasks of a File System.....	164
9.5.3 Obtaining Details About a File System Asynchronous Task.....	169
9.5.4 Canceling or Deleting an Asynchronous Task of a File System.....	173
9.6 Storage Interworking Management.....	176
9.6.1 Adding a Backend Target.....	176
9.6.2 Querying Backend Targets.....	189
9.6.3 Obtaining Details About a Backend Target.....	196
9.6.4 Deleting a Backend Target.....	202
9.6.5 Updating the Properties of a Storage Backend.....	205
9.6.6 Updating the Auto Synchronization Policy of a Storage Backend.....	212
9.6.7 Creating an Import or Export Task.....	217
9.6.8 Querying Details About an Import or Export Task.....	224
9.6.9 Listing Import and Export Tasks.....	228
9.6.10 Deleting an Import or Export Task.....	231
9.6.11 Updating a File System.....	234
9.7 Directory Management.....	236
9.7.1 Creating Quotas for a Directory.....	236
9.7.2 Updating Quotas of a Directory.....	240
9.7.3 Querying Quotas of a Directory.....	243
9.7.4 Deleting Quotas of a Directory.....	246
9.7.5 Creating a Directory.....	248
9.7.6 Checking Whether a Directory Exists.....	251
9.7.7 Deleting a Directory.....	255
9.7.8 Querying the Usage of a File System Directory.....	257
9.8 Permissions Management.....	260
9.8.1 Creating a Permission Rule.....	260
9.8.2 Querying Permission Rules of a File System.....	265
9.8.3 Querying a Permission Rule of a File System.....	267
9.8.4 Modifying a Permission Rule.....	270
9.8.5 Deleting a Permissions Rule.....	274
9.8.6 Creating and Binding the LDAP Configuration.....	276
9.8.7 Querying the LDAP Configuration.....	280
9.8.8 Modifying the LDAP Configuration.....	282
9.8.9 Deleting the LDAP Configuration.....	286
9.9 Task Management.....	288
9.9.1 Querying Details About a Task.....	288
10 General Purpose File System APIs.....	293
10.1 File Systems.....	293
10.1.1 Creating a File System.....	293
10.1.2 Deleting a File System.....	295
10.1.3 Listing File Systems.....	296

10.2 File System Access Rules.....	298
10.2.1 Configuring a File System ACL.....	298
10.2.2 Obtaining File System ACL Information.....	300
10.2.3 Deleting a File System ACL.....	302
10.3 Tags.....	303
10.3.1 Batch Adding Tags to a Resource.....	304
10.3.2 Batch Deleting Tags from a Resource.....	307
10.3.3 Querying Tags of a Resource.....	308
10.3.4 Listing Resources.....	310
10.3.5 Querying the Number of Resources.....	316
10.3.6 Querying Tags by Project.....	319
11 Permissions Policies and Supported Actions.....	322
11.1 Introduction.....	322
11.2 Supported Actions.....	323
11.2.1 SFS Actions.....	323
11.2.2 SFS Turbo Actions.....	329
12 Common Parameters.....	333
12.1 SFS Turbo File System Statuses.....	333
12.2 SFS Turbo File System Substatuses.....	333
13 Appendix.....	334
13.1 Status Codes.....	334
13.2 SFS Turbo Error Codes.....	335
13.3 General Purpose File System Error Codes.....	340
13.4 Obtaining Access Keys (AK/SK).....	347
13.5 Obtaining a Project ID.....	348

1 Before You Start

1.1 Overview

Welcome to *Scalable File Service API Reference*. Scalable File Service (SFS) is a network attached storage (NAS) service that provides scalable, high-performance file storage. With SFS, you can enjoy shared file access spanning multiple Elastic Cloud Servers (ECSs), Bare Metal Servers (BMSs), and containers created on Cloud Container Engine (CCE).

This document describes how to use application programming interfaces (APIs) to perform operations on SFS resources, such as creating, querying, deleting, and updating a file system. For details about all supported operations, see [API Overview](#).

If you plan to access SFS through an API, ensure that you are familiar with SFS concepts. For details, see [Service Overview](#).

1.2 API Calling

SFS supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS requests. For details about API calling, see [Calling APIs](#).

1.3 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoint of SFS, see [Regions and Endpoints](#).

1.4 Constraints

- The numbers of file systems that you can create and their capacities are determined by your quotas. To view or increase the quotas, see [Quotas](#).
- For more constraints, see API description.
- General Purpose File System APIs can only be called on a private network, not a public network.

1.5 Concepts

- **Account**

An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity, which should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.
- **User**

An IAM user is created by an account in IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys).

API authentication requires information such as the account name, username, and password.
- **Region**

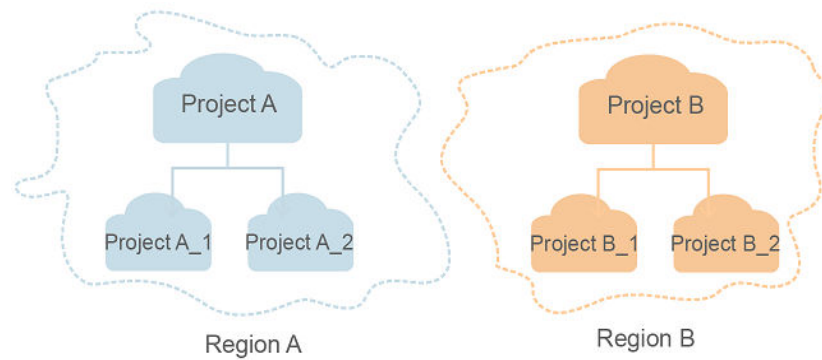
Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

For details, see [Region and AZ](#).
- **AZ**

An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.
- **Project**

A project corresponds to a region. Default projects are defined to group and physically isolate resources (including computing, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolation model



- **Enterprise project**
Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.
For details about enterprise projects and about how to obtain enterprise project IDs, see [Enterprise Management User Guide](#).

2 API Overview

APIs provided by SFS include SFS APIs, SFS Turbo APIs, and native OpenStack APIs.

These APIs allow you to use all SFS functions.

The call frequency limit of SFS Capacity-Oriented APIs is 400 calls/minute. If you call a large number of APIs at a time, some APIs may fail to be called. You are advised to evenly arrange API calls.

Some SFS Capacity-Oriented APIs are the same as those provided by OpenStack Manila of the Mitaka version. For details about how to use them, visit the community:

<https://docs.openstack.org/api-ref/shared-file-system/>

For details about whether an SFS Capacity-Oriented API supports Enterprise Project, see [API Permissions](#).

If the description about an API in this document differs from that in the community, the description in this document is used.

SFS Capacity-Oriented APIs

Table 2-1 API overview

File System Type	Type	Category	Description
SFS	SFS API	Tag management	You can use APIs in this category to tag your shared file system, making them easier to be managed.
SFS	Native OpenStack API	API version queries	You can use APIs in this category to query the versions and details of all APIs.
		File systems	You can use APIs in this category to create shared file systems and obtain detailed information about them, such as the shared paths.

File System Type	Type	Category	Description
		File system access rules	You can use APIs in this category to add, modify, and delete file system access rules, such as configuring VPC.
		Limits management	If the number of created shared file systems reaches the upper limit, you can increase quota by using APIs in this category.
		Capacity expansion and reduction	If you want to change the capacity of a created shared file system, you can use APIs in this category to expand or reduce the capacity.

General Purpose File System APIs

Table 2-2 API overview

Type	Category	Description
General Purpose File System API	File systems	You can use APIs in this category to create, delete, or list General Purpose File Systems.
General Purpose File System API	File system access rules	You can use APIs in this category to configure, obtain, or delete file system access rules.
General Purpose File System API	Tags	You can use APIs in this category to add, delete, or query tags in batches.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for **obtaining a user token** as an example to demonstrate how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

Table 3-1 URI parameter description

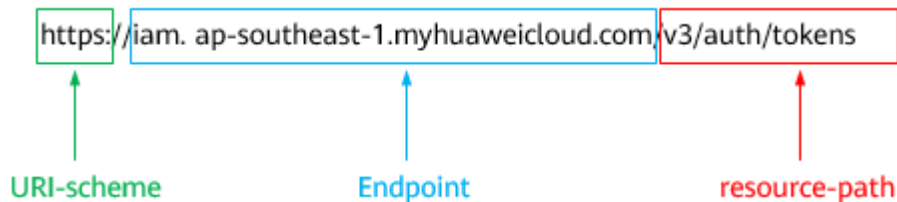
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints . For example, the endpoint of IAM in region CN-Hong Kong is iam.ap-southeast-1.myhuaweicloud.com .
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of <i>Parameter name=Parameter value</i> . For example, ? limit=10 indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **CN-Hong Kong** region, obtain the endpoint of IAM (iam.ap-southeast-1.myhuaweicloud.com) for this region and the resource-path (/v3/auth/tokens) in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

```
https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server.

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.

Method	Description
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to [obtain a user token](#), the request method is **POST**. The request is as follows:

POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of <i>Hostname:Port number</i> . If the port number is not specified, the default port is used. The default port number for https is 443 .	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value application/json is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in Obtaining a Project ID .	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	Specifies the user token. It is a response to the API for obtaining a user token (This is the only API that does not require authentication). After the request is processed, the value of X-Subject-Token in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZlhvcNAQcCo...ggg1BBIINPXsidG9rZ

 **NOTE**

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added in the request.

For more details, see "Authentication Using AK/SK" in [Authentication](#).

The API used to [obtain a user token](#) does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format (for example, JSON or XML) as specified in the **Content-Type** header field. The request body transfers content except the request header.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to [obtain a user token](#), the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*,

domainname, ******* (login password), and *xxxxxxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from [Regions and Endpoints](#).

 **NOTE**

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see [Obtaining a User Token](#).

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

If all data required for the API request is available, you can send the request to call the API through [curl](#), [Postman](#), or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token authentication: Requests are authenticated using tokens.
- AK/SK authentication: Requests are encrypted using AK/SK pairs. AK/SK authentication is recommended because it is more secure than token authentication.

Token Authentication

 **NOTE**

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API. You can obtain a token by calling the [Obtaining User Token](#) API.

IMS is a project-level service. When you call the API, set **auth.scope** in the request body to **project**.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username", // IAM user name
          "password": "*****", // IAM user password
          "domain": {
            "name": "domainname" // Name of the account to which the IAM user belongs
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxx" // Project name
      }
    }
  }
}
```

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK Authentication

NOTE

AK/SK authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token authentication is recommended.

In AK/SK authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key, which is used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or using the signing SDK. For details about how to sign requests and use the signing SDK, see [API Request Signing Guide](#).

NOTE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see [Status Codes](#).

For example, if status code **201** is returned for calling the API used to [obtain a user token](#), the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

[Figure 3-2](#) shows the response header fields for the API used to [obtain a user token](#). The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 Header fields of the response to the request for obtaining a user token

```

connection → keep-alive

content-type → application/json

date → Tue, 12 Feb 2019 06:52:13 GMT

server → Web Server

strict-transport-security → max-age=31536000; includeSubdomains;

transfer-encoding → chunked

via → proxy A

x-content-type-options → nosniff

x-download-options → noopen

x-frame-options → SAMEORIGIN

x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5

x-subject-token
→ MIIVXQVJKoZIhvcNAQcCoIIYJCCEoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOnsiZXhwaXJlc19hdCI6ijlwMTktMDItMTNUMC
fj3KIs6YgKnpVNRbW2eZ5eb78SZ0kqjACgkIQ1wi4JIGzrpd18LGXK5tdfq4lqHCYb8P4NaY0NyejcAgzJVeFYtLWT1.GSO0zxKZmlQHQj82HBqHdglZO9fuEbL5dMhdavj+33wEI
xHRC9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jqglFkNPQuFSOUB+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUUpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection → 1; mode=block;

```

(Optional) Response Body

The body of a response is often returned in a structured format (for example, JSON or XML) as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to [obtain a user token](#).

```
{
  "token": {
```

```
"expires_at": "2019-02-13T06:52:13.855000Z",  
"methods": [  
  "password"  
],  
"catalog": [  
  {  
    "endpoints": [  
      {  
        "region_id": "az-01",  
.....
```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{  
  "error_msg": "The format of message is error",  
  "error_code": "AS.0001"  
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 Calling General Purpose File System APIs

4.1 Constructing a Request

This section describes the structure of a REST API request.

Request URI

SFS uses URI to locate specific file systems and their parameters. Use URIs when you want to operate resources.

The following provides a common URI format. The parameters in square brackets [] are optional.

protocol://[filesystem.]domain[:port]/[?param]

Table 4-1 URI parameters

Parameter	Description	Mandatory
protocol	Protocol used for sending requests, which can be either HTTP or HTTPS. HTTPS is a protocol that ensures secure access to resources. SFS supports both HTTP and HTTPS.	Yes
filesystem	Resource path of a file system, identifying only one file system in SFS	No
domain	Domain name or IP address of the server for saving resources	Yes

Parameter	Description	Mandatory
port	Port enabled for protocols used for sending requests. The value varies with software server deployment. If no port number is specified, the protocol uses the default value. Each transmission protocol has its default port number. For example, HTTP uses port number 80 and HTTPS uses port number 443 by default. In SFS, HTTP port number is 80 and that of HTTPS is 443 .	No
param	A specific resource contained by a file system. Default value of this parameter indicates that the file system itself is obtained.	No

NOTICE

All API requests except those for the file system list must contain the file system name. Based on the DNS resolution performance and reliability, SFS requires that the file system name must be placed in front of the **domain** when a request carrying a file system name is constructed to form a third-level domain name, also mentioned as virtual hosting access domain name.

Request Method

HTTP methods, which are also called operations or actions, specify the type of operations that you are requesting.

Table 4-2 HTTP request methods supported by SFS

Method	Description
GET	Requests the server to return specific resources, for example, to list file systems.
PUT	Requests the server to update specific resources, for example, creating file systems.
POST	Requests a server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, file systems.
HEAD	Same as GET except that the server must return only the response header.
OPTIONS	Requests the server to check whether the user has the permissions to operate a resource.

Request Headers

Refers to optional and additional request fields, for example a field required by a specific URI or HTTP method. [Table 4-3](#) describes some common request header fields.

Table 4-3 Common request headers

Header	Description	Mandatory
Authorization	Signature information contained in a request message Type: string No default value. Conditional: optional for anonymous requests and required for other requests.	Conditionally required
Content-Length	The message length (excluding headers) defined in RFC 2616 Type: string No default value. Conditional: required for PUT requests and those requests that load XML content.	Conditionally required
Content-Type	The content type of the requested resource, for example, text/plain Type: string No default value.	No
Date	The time when a request is initiated, for example, Wed, 27 Jun 2018 13:39:15 +0000 . Type: string No default value. Conditional: optional for anonymous requests or those requests containing header x-obs-date , required for other requests.	Conditionally required
Host	The host address, for example, filesystem.sfs3.region.myhuaweicloud.com . Type: string No default value.	Yes

(Optional) Request Body

A request body is generally sent in a structured format (for example, JSON or XML). It corresponds to **Content-Type** in the request header and is used to transfer content other than the request header. If the request body contains Chinese characters, these characters must be coded in UTF-8.

The request body varies according to the APIs. Certain APIs do not require the request body, such as the GET and DELETE APIs.

Sending a Request

There are two methods to initiate requests based on the constructed request messages:

- cURL
cURL is a command-line tool used to perform URL operations and transmit information. cURL acts as an HTTP client that can send HTTP requests to the server and receive response messages. cURL is applicable to API debugging. For more information about cURL, visit <https://curl.haxx.se/>. cURL cannot calculate signatures. When cURL is used, only anonymous public SFS resources can be accessed.
- Coding
You can use code to make API calls, and to assemble, send, and process request messages. It can be implemented by coding.

4.2 Authentication

4.2.1 User Signature Authentication

SFS signs a request using AK/SK. When a client is sending a request to SFS, the message header must contain the SK, request time, request type, and other information of the signature.

- AK: access key ID, which is a unique identifier associated with a secret access key (SK). The AK and SK are used together to obtain an encrypted signature for a request. Format example: **HCY8BGCN1YM5ZWYOK1MH**
- SK: secret access key, which is used together with the AK to sign requests, identify a request sender, and prevent the request from being modified. Format example: **9zYwf1uabSQY0JTnFqbUqG7vcfqYBaTdXde2GUcq**

A user can obtain the AK and SK from IAM. For details, see [Obtaining Access Keys \(AK/SK\)](#).

SFS provides the signature calculation method based on the application scenario [Authentication of Signature in a Header](#).

[Table 4-4](#) shows the user signature verification process in which a signature is carried in a header. For details about the parameters and code examples of authentication of signature in a header, see [Authentication of Signature in a Header](#).

Table 4-4 Signature calculation and verification procedure

Procedure		Example
Signature calculation	1. Construct an HTTP message.	PUT /HTTP/1.1 Host: filesystem.sfs3.region.myhuaweicloud.com Date: Tue, 04 Jun 2019 06:54:59 GMT Content-Type: text/plain Content-Length: 5913
	2. Calculate StringToSign based on the signature rule.	StringToSign = HTTP-Verb + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Date + "\n" + CanonicalizedHeaders + CanonicalizedResource
	3. Prepare the AK and SK.	AK: ***** SK: *****
	4. Calculate Signature .	Signature = Base64(HMAC-SHA1(SecretAccessKeyID , UTF-8-Encoding-Of(StringToSign)))
	5. Add a signature header and send the request to SFS.	PUT /object HTTP/1.1 Host: filesystem.sfs3.region.myhuaweicloud.com Date: Tue, 04 Jun 2019 06:54:59 GMT Content-Type: text/plain Content-Length: 5913 Authorization: OBS AccessKeyID:Signature
Signature authentication	6. Receive the HTTP message.	PUT / HTTP/1.1 Host: filesystem.sfs3.region.myhuaweicloud.com Date: Tue, 04 Jun 2019 06:54:59 GMT Content-Type: text/plain Content-Length: 5913 Authorization: OBS AccessKeyID:Signature
	7. Obtain the SK based on the AK in the request.	Obtain the AK from the authorization header and obtain the SK of the user from IAM.
	8. Calculate StringToSign based on the signature rule.	StringToSign = HTTP-Verb + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Date + "\n" + CanonicalizedHeaders + CanonicalizedResource
	9. Calculate Signature .	Signature = Base64(HMAC-SHA1(SecretAccessKeyID , UTF-8-Encoding-Of(StringToSign)))

Procedure		Example
	10. Authenticate the signature.	<p>Verify that the value of Signature in the authorization header is the same as the value of Signature calculated by the server.</p> <p>If the two values are the same, the signature verification is successful.</p> <p>If the two values are different, the signature verification fails.</p>

4.2.2 Authentication of Signature in a Header

For all General Purpose File System API operations, the identity authentication can be done by carrying signatures in headers.

In the header, the signature is carried in the authorization header field of the HTTP message. The format of the message header is as follows:

```
Authorization: OBS AccessKeyID:signature
```

The signature algorithm process is as follows:

1. Construct the request character string (StringToSign).
2. Perform UTF-8 encoding on the result obtained from the preceding step.
3. Use the SK to perform the HMAC-SHA1 signature calculation on the result obtained from step 2.
4. Perform Base64 encoding on the result of step 3 to obtain the signature.

The StringToSign is constructed according to the following rules. [Table 4-5](#) describes the parameters.

```
StringToSign =
  HTTP-Verb + "\n" +
  Content-MD5 + "\n" +
  Content-Type + "\n" +
  Date + "\n" +
  CanonicalizedHeaders + CanonicalizedResource
```

Table 4-5 Parameters required for constructing a StringToSign

Parameter	Description
HTTP-Verb	An HTTP request method supported by the REST API. The value can be an HTTP verb such as PUT , GET , or DELETE .
Content-MD5	The base64-encoded 128-bit MD5 digest of the message according to RFC 1864. This parameter can be empty.
Content-Type	The message type, for example, text/plain . If a request does not contain this header field, this parameter will be processed as an empty string.

Parameter	Description
Date	<p>The time when a request is initiated. This parameter uses the RFC 1123 time format. If the deviation between the time specified by this parameter and the server time is over 15 minutes, the server returns error 403.</p> <p>This parameter is an empty string when the x-obs-date is specified.</p>
CanonicalizedHeaders	<p>The SFS request header field in an HTTP request header, referring to header fields started with x-obs-, for example, x-obs-date, x-obs-acl, and x-obs-meta-*.</p> <ol style="list-style-type: none"> 1. All characters of keywords in a request header field must be converted to lowercase letters (content values must be case sensitive, for example, x-obs-storage-class:STANDARD). If a request contains multiple header fields, these fields should be organized by keyword in the alphabetical order from a to z. 2. If multiple header fields in a request have the same prefix, combine the header fields into one. For example, x-obs-meta-name:name1 and x-obs-meta-name:name2 should be reorganized into x-obs-meta-name:name1,name2. Use comma to separate the values. 3. Keywords in the request header field cannot contain non-ASCII or unrecognizable characters, which are also not advisable for values in the request header field. If the two types of characters are necessary, they should be encoded and decoded on the client side. Either URL encoding or Base64 encoding is acceptable, but the server does not perform decoding. 4. Delete meaningless spaces and tabs in a header field. For example, x-obs-meta-name: name (with a meaningless space in the front of name) must be changed to x-obs-meta-name:name. 5. Each header field occupies a separate line.

Parameter	Description
CanonicalizedResource	<p>The SFS resource specified by an HTTP request. This parameter is constructed as follows:</p> <p><File system name + Object name> + [Subresource 1] + [Subresource 2] + ...</p> <ol style="list-style-type: none"> 1. File system name and object name, for example, /filesystem/object. If no object name is specified, the entire file system is listed, for example, /filesystem/. If file system name is not specified either, the value of this field is /. 2. If a subresource (such as ?sfsacl) exists, the subresource must be added. 3. If there are multiple subresources, sort them in the alphabetical order from a to z, and use & to combine the subresources. <p>NOTE</p> <ul style="list-style-type: none"> • A subresource is unique. Do not add subresources with the same keyword (for example, key=value1&key=value2) in the same request URL. In this case, signature is computed only based on the first subresource, and only the value of the first subresource takes effect on the actual service.

The following table provides some examples of generating StringToSign.

Table 4-6 Obtaining the ACL of a file system

Request Header	StringToSign
GET /?sfsacl HTTP/1.1 Host: filesystem.sfs3.region.myhuaweicloud.com Date: Sat, 12 Oct 2015 08:12:38 GMT	GET \n \n \n Sat, 12 Oct 2015 08:12:38 GMT\n /filesystem/?sfsacl

Content-MD5 Algorithm in Java

```
import java.security.MessageDigest;
import sun.misc.BASE64Encoder;
import java.io.UnsupportedEncodingException;
import java.security.NoSuchAlgorithmException;

public class Md5{
    public static void main(String[] args) {
        try {
            String exampleString = "blog";
            MessageDigest messageDigest = MessageDigest.getInstance("MD5");
            BASE64Encoder encoder = new BASE64Encoder();
            String contentMd5 = encoder.encode(messageDigest.digest(exampleString.getBytes("utf-8")));
            System.out.println("Content-MD5:" + contentMd5);
        } catch (NoSuchAlgorithmException | UnsupportedEncodingException e)
    }
}
```

```
    {  
        e.printStackTrace();  
    }  
}
```

The signature is generated as follows based on the `StringToSign` and `SK`. The hash-based message authentication code algorithm (HMAC algorithm) is used to generate the signature.

```
Signature = Base64( HMAC-SHA1( YourSecretAccessKeyID, UTF-8-Encoding-Of( StringToSign ) ) )
```

For example, to create a file system named **newfilesystem2** in a region, the client request format is as follows:

```
PUT / HTTP/1.1  
Host: newfilesystem2.sfs3.region.myhuaweicloud.com  
Content-Length: length  
Date: Fri, 06 Jul 2018 03:45:51 GMT  
x-obs-acl:private  
x-obs-storage-class:STANDARD  
Authorization: OBS UDSIAMSTUBTEST000254:ydH8ffpcbS6YpeOMcEZfn0wE90c=  
  
<CreateBucketConfiguration xmlns="http://obs.myhwclouds.com/doc/2015-06-30/">  
  <Location>region</Location>  
</CreateBucketConfiguration>
```

Signature Algorithm in Java

```
import java.io.UnsupportedEncodingException;  
import java.net.URLEncoder;  
import java.security.InvalidKeyException;  
import java.security.NoSuchAlgorithmException;  
import java.util.ArrayList;  
import java.util.Arrays;  
import java.util.Base64;  
import java.util.Collections;  
import java.util.HashMap;  
import java.util.List;  
import java.util.Locale;  
import java.util.Map;  
import java.util.TreeMap;  
  
import javax.crypto.Mac;  
import javax.crypto.spec.SecretKeySpec;  
  
import org.omg.CosNaming.IstringHelper;  
  
public class SignDemo {  
  
    private static final String SIGN_SEP = "\n";  
  
    private static final String SFS_PREFIX = "x-obs-";  
  
    private static final String DEFAULT_ENCODING = "UTF-8";  
  
    private static final List<String> SUB_RESOURCES = Collections.unmodifiableList(Arrays.asList(  
        "CDNNotifyConfiguration", "acl", "append", "atname", "backtosource", "cors", "customdomain",  
"delete",  
        "deletebucket", "directcoldaccess", "encryption", "inventory", "length", "lifecycle", "location",  
"logging",  
        "metadata", "modify", "name", "notification", "orchestration", "partNumber", "policy", "position",  
"quota",  
        "rename", "replication", "requestPayment", "response-cache-control", "response-content-  
disposition",  
        "response-content-encoding", "response-content-language", "response-content-type", "response-  
expires",
```

```
"restore", "select", "storageClass", "storagePolicy", "storageinfo", "tagging", "torrent", "truncate",
"uploadId", "uploads", "versionId", "versioning", "versions", "website", "x-image-process",
"x-image-save-bucket", "x-image-save-object", "x-obs-security-token"));

private String ak;

private String sk;

public String urlEncode(String input) throws UnsupportedEncodingException
{
    return URLEncoder.encode(input, DEFAULT_ENCODING)
        .replaceAll("%7E", "~") //for browser
        .replaceAll("%2F", "/")
        .replaceAll("%20", "+");
}

private String join(List<?> items, String delimiter)
{
    StringBuilder sb = new StringBuilder();
    for (int i = 0; i < items.size(); i++)
    {
        String item = items.get(i).toString();
        sb.append(item);
        if (i < items.size() - 1)
        {
            sb.append(delimiter);
        }
    }
    return sb.toString();
}

private boolean isValid(String input) {
    return input != null && !input.equals("");
}

public String hamcSha1(String input) throws NoSuchAlgorithmException, InvalidKeyException,
UnsupportedEncodingException {
    SecretKeySpec signingKey = new SecretKeySpec(this.sk.getBytes(DEFAULT_ENCODING), "HmacSHA1");
    Mac mac = Mac.getInstance("HmacSHA1");
    mac.init(signingKey);
    return Base64.getEncoder().encodeToString(mac.doFinal(input.getBytes(DEFAULT_ENCODING)));
}

private String stringToSign(String httpMethod, Map<String, String[]> headers, Map<String, String>
queries,
    String bucketName, String objectName) throws Exception{
    String contentMd5 = "";
    String contentType = "";
    String date = "";

    TreeMap<String, String> canonicalizedHeaders = new TreeMap<String, String>();

    String key;
    List<String> temp = new ArrayList<String>();
    for(Map.Entry<String, String[]> entry : headers.entrySet()) {
        key = entry.getKey();
        if(key == null || entry.getValue() == null || entry.getValue().length == 0) {
            continue;
        }

        key = key.trim().toLowerCase(Locale.ENGLISH);
        if(key.equals("content-md5")) {
            contentMd5 = entry.getValue()[0];
            continue;
        }

        if(key.equals("content-type")) {
            contentType = entry.getValue()[0];
            continue;
        }
    }
}
```

```
    }

    if(key.equals("date")) {
        date = entry.getValue()[0];
        continue;
    }

    if(key.startsWith(OBS_PREFIX)) {

        for(String value : entry.getValue()) {
            if(value != null) {
                temp.add(value.trim());
            }
        }
        canonicalizedHeaders.put(key, this.join(temp, ","));
        temp.clear();
    }
}

if(canonicalizedHeaders.containsKey("x-obs-date")) {
    date = "";
}

// handle method/content-md5/content-type/date
StringBuilder stringToSign = new StringBuilder();
stringToSign.append(httpMethod).append(SIGN_SEP)
    .append(contentMd5).append(SIGN_SEP)
    .append(contentType).append(SIGN_SEP)
    .append(date).append(SIGN_SEP);

// handle canonicalizedHeaders
for(Map.Entry<String, String> entry : canonicalizedHeaders.entrySet()) {
    stringToSign.append(entry.getKey()).append(":").append(entry.getValue()).append(SIGN_SEP);
}

// handle CanonicalizedResource
stringToSign.append("/");
if(this.isValid(bucketName)) {
    stringToSign.append(bucketName).append("/");
    if(this.isValid(objectName)) {
        stringToSign.append(this.urlEncode(objectName));
    }
}

TreeMap<String, String> canonicalizedResource = new TreeMap<String, String>();
for(Map.Entry<String, String> entry : queries.entrySet()) {
    key = entry.getKey();
    if(key == null) {
        continue;
    }

    if(SUB_RESOURCES.contains(key)) {
        canonicalizedResource.put(key, entry.getValue());
    }
}

if(canonicalizedResource.size() > 0) {
    stringToSign.append("?");
    for(Map.Entry<String, String> entry : canonicalizedResource.entrySet()) {
        stringToSign.append(entry.getKey());
        if(this.isValid(entry.getValue())) {
            stringToSign.append("=").append(entry.getValue());
        }
    }
}

// System.out.println(String.format("StringToSign:%s%s", SIGN_SEP, stringToSign.toString()));
```

```
        return stringToSign.toString();
    }

    public String headerSignature(String httpMethod, Map<String, String[]> headers, Map<String, String>
queries,
        String bucketName, String objectName) throws Exception {

        //1. stringToSign
        String stringToSign = this.stringToSign(httpMethod, headers, queries, bucketName, objectName);

        //2. signature
        return String.format("OBS %s:%s", this.ak, this.hamcSha1(stringToSign));
    }

    public String querySignature(String httpMethod, Map<String, String[]> headers, Map<String, String>
queries,
        String bucketName, String objectName, long expires) throws Exception {
        if(headers.containsKey("x-obs-date")) {
            headers.put("x-obs-date", new String[] {String.valueOf(expires)});
        }else {
            headers.put("date", new String[] {String.valueOf(expires)});
        }
        //1. stringToSign
        String stringToSign = this.stringToSign(httpMethod, headers, queries, bucketName, objectName);

        //2. signature
        return this.urlEncode(this.hamcSha1(stringToSign));
    }

    public static void main(String[] args) throws Exception {

        SignDemo demo = new SignDemo();
        demo.ak = "<your-access-key-id>";
        demo.sk = "<your-secret-key-id>";

        String bucketName = "bucket-test";
        String objectName = "hello.jpg";
        Map<String, String[]> headers = new HashMap<String, String[]>();
        headers.put("date", new String[] {"Sat, 12 Oct 2015 08:12:38 GMT"});
        headers.put("x-obs-acl", new String[] {"private"});
        Map<String, String> queries = new HashMap<String, String>();
        queries.put("acl", null);

        System.out.println(demo.headerSignature("PUT", headers, queries, bucketName, objectName));
    }
}
```

The calculation result of the signature is as follows (it varies with the execution time): YdH8ffpcbS6YpeOMcEZfn0wE90c=

Signature Algorithm in Python

```
import sys
import os
import hashlib
import hmac
import binascii
from datetime import datetime
IS_PYTHON2 = sys.version_info.major == 2 or sys.version < '3'
"""Hard-coded or plaintext SecretAccessKeyID is risky. For security purposes, encrypt your access key, store
it in the configuration file or environment variables, and decrypt it before using it. In this example,
SecretAccessKeyID is stored in the environment variables for identity authentication. Before running the
code in this example, configure environment variable SECRET_ACCESS_KEY_ID."""
yourSecretAccessKeyID = os.environ.get('SECRET_ACCESS_KEY_ID')
httpMethod = "PUT"
contentType = "application/xml"
httpMethod = "PUT"
```

```
contentType = "application/xml"
# "date" is the time when the request was actually generated
date = datetime.utcnow().strftime('%a, %d %b %Y %H:%M:%S GMT')
canonicalizedHeaders = "x-obs-acl:private\n"
CanonicalizedResource = "/newfilesystem2"
canonical_string = httpMethod + "\n" + "\n" + contentType + "\n" + date + "\n" + canonicalizedHeaders + CanonicalizedResource
if IS_PYTHON2:
    hashed = hmac.new(yourSecretAccessKeyID, canonical_string, hashlib.sha1)
    encode_canonical = binascii.b2a_base64(hashed.digest())[:-1]
else:
    hashed = hmac.new(yourSecretAccessKeyID.encode('UTF-8'),
canonical_string.encode('UTF-8'),hashlib.sha1)
    encode_canonical = binascii.b2a_base64(hashed.digest())[:-1].decode('UTF-8')
print encode_canonical
```

The calculation result of the signature is as follows (it varies with the execution time): YdH8ffpcbS6YpeOMcEZfn0wE90c=

Signature Algorithm in the C Programming Language

Download the sample code for calculating the signature in the C programming language.

1. The API for calculating the signature is contained in the **sign.h** header file.
2. The sample code for calculating the signature is contained in the **main.c** header file.

Signature Mismatch Error Handling

During an SFS API call, if the following error is reported,

Status code: 403 Forbidden

Error code: SignatureDoesNotMatch

Error message: The request signature we calculated does not match the signature you provided. Check your key and signing method.

Contact technical support.

4.2.3 Signature Generators

SFS offers visualized tools for you to easily generate signatures.

Table 4-7 Signature generators

Calculation Method	How to Obtain
Authenticating the signature in a header	Visit Generate Header .

During an SFS API call, if the following error is reported,

Status code: 403 Forbidden

Error code: SignatureDoesNotMatch

Error message: The request signature we calculated does not match the signature you provided. Check your key and signing method.

Contact technical support.

4.3 Returned Values

After sending a request, you will receive a response, including the status code, response header, and response body.

Status Codes

A status code is a group of digits indicating the status of a response. It ranges from 2xx (indicating successes) to 4xx or 5xx (indicating errors). For more information, see [Status Codes](#).

Response Headers

A response header corresponds to a request header, for example, Content-Type.

For details about common response headers, see [Table 4-8](#).

Table 4-8 Common response headers

Header	Description
Content-Length	The length (in bytes) of the response body. Type: string Default value: none
Connection	Indicates whether the connection to the server is a long connection or a short connection. Type: string Valid values: keep-alive close Default value: none
Date	The time when a response is returned. Type: string Default value: none
x-obs-id-2	A special symbol that helps troubleshoot faults. Type: string Default value: none
x-obs-request-id	The unique identifier of the request. The value is generated by the SFS service and can be used for troubleshooting. Type: string Default value: none

(Optional) Response Body

A response body is generally returned in a structured format (for example, JSON or XML), corresponding to **Content-Type** in the response header, and is used to transfer content other than the response header.

5 Getting Started (SFS Capacity-Oriented)

Scenarios

SFS provides high-performance network-attached storage (NAS) that is scalable on demand. A shared file system can be shared with multiple Elastic Cloud Servers (ECSs) and Bare Metal Servers (BMSs). If you need a fully hosted shared file storage and want to access a file system on multiple ECSs, SFS is perfect for you.

The following describes how to call the API for [Creating a Shared File System](#). For details, see [Making an API Request](#).

Prerequisites

You need to plan the region where a file system resides and determine the endpoint for calling an API based on the region. It can be obtained from [Regions and Endpoints](#).

Creating a Shared File System

The following is the sample code about how to create a shared file system with the simplest configurations:

```
{
  "share": {
    "description": "test description",
    "share_type": "default",
    "name": "share_London",
    "metadata": {
      "key1": "value1",
      "key2": "value2"
    },
    "share_proto": "NFS",
    "size": 10,
    "is_public": false
  }
}
```

- **description:** Specifies the description of the shared file system, which adds remarks to the shared file system.
- **share_type:** Specifies the name of a share type. A share type is used to specify the type of the storage service to be allocated.
- **share_proto:** Specifies the protocol types of the shared file system.

- **name**: Specifies the custom name of the shared file system. For example, **share_London**.
- **size**: Specifies the size (in GB) of the shared file system.
- **is_public**: Specifies the visibility level of the shared file system. If it is set to **true**, the file system can be seen publicly. If it is set to **false**, the file system can be seen privately. The default value is **false**.
- **metadata**: Specifies the metadata information of the shared file system. The value consists of one or more key and value pairs organized as a dictionary of strings.

Creating an Encrypted Shared File System

You can also encrypt a shared file system. You only need to add parameters related to encryption of a shared file system to the metadata of the request body. The following is an example:

```
{
  "share": {
    "share_type": null,
    "name": "test",
    "snapshot_id": null,
    "description": "test description",
    "metadata": {
      "#sfs_crypt_key_id": "9130c90d-73b8-4203-b790-d49f98d503df",
      "#sfs_crypt_domain_id": "3b2d9670690444c582942801ed7d457b",
      "#sfs_crypt_alias": "sfs/default"
    },
    "share_proto": "NFS",
    "share_network_id": null,
    "size": 1,
    "is_public": false
  }
}
```

- **#sfs_crypt_key_id**: Specifies the encryption key ID. If **#sfs_crypt_key_id**, **#sfs_crypt_domain_id**, and **#sfs_crypt_alias** exist at the same time, the data encryption function is enabled.
- **#sfs_crypt_domain_id**: Specifies the tenant domain ID. If **#sfs_crypt_domain_id**, **#sfs_crypt_key_id**, and **#sfs_crypt_alias** exist at the same time, the data encryption function is enabled.
- **#sfs_crypt_alias**: Specifies the encryption key alias. If **#sfs_crypt_alias**, **#sfs_crypt_key_id**, and **#sfs_crypt_domain_id** exist at the same time, the data encryption function is enabled.

6 Getting Started with SFS Turbo

This section describes how to use APIs by calling an API to create an SFS Turbo file system.

NOTE

The token obtained from IAM is valid for only 24 hours. If you want to use one token for authentication, you can cache it to avoid frequently calling.

Involved APIs

If you use a token for authentication, you must obtain the token and add **X-Auth-Token** to the request header of the API when making a call. The following APIs are involved in the request for creating an SFS Turbo file system:

- API for obtaining tokens from IAM
- For details about how to create an SFS Turbo file system, see [Creating a File System](#).

Procedure

Step 1 Obtain the token by following instructions in [Authentication](#).

Step 2 Add **X-Auth-Token** to the request header.

Step 3 Specify the following parameters in the request body:

```
{
  "share": {
    "name": "sfs-turbo-test",
    "share_proto": "NFS",
    "share_type": "STANDARD",
    "size": 100,
    "availability_zone": "az1",
    "vpc_id": "d651ea2b-2b20-4c6d-8bbf-2adcec18dac9",
    "subnet_id": "b8884abe-f47b-4917-9f6c-f64825c365db",
    "security_group_id": "8c4ebbd0-6edf-4aae-8353-81ce6d06e1f4"
  }
}
```

Step 4 Send the request **POST** https://Endpoint of SFS Turbo/v1/{project_id}/sfs-turbo/shares.

Step 5 After the request is successfully responded, the ID and name of the SFS Turbo file system are returned.

If the request fails, an error code and error information are returned. For details about the error codes, see the abnormal return values of the corresponding API.

Query SFS Turbo file system details based on the returned file system ID. For details, see [Querying Details About a File System](#).

If the returned status of the file system is **200**, the SFS Turbo file system is successfully created. For details about the return values of request exceptions, see the abnormal return values of the corresponding API. For other statuses, see [SFS Turbo File System Statuses](#).

You can query and delete an SFS Turbo file system based on the file system ID.

----End

Configuration Example

If the token has been obtained, you can run the following **curl** command to create an SFS Turbo file system:

```
curl -k -i -X POST -H "X-Auth-Token: token_value" -H "Content-Type: application/json" -d '{"share": {"name": "sfs-turbo-test", "share_proto": "NFS", "share_type": "STANDARD", "size": 100, "availability_zone": "az1", "vpc_id": "d651ea2b-2b20-4c6d-8bbf-2adcec18dac9", "subnet_id": "b8884abe-f47b-4917-9f6c-f64825c365db", "security_group_id": "8c4ebbd0-6edf-4aae-8353-81ce6d06e1f4"}}' "https://127.0.0.1:8979/v1/xxx/bx/bex5cfx41f0a08ay915fd79240d/sfs-turbo/shares"
```

7 Getting Started with General Purpose File System

7.1 Creating a File System

Scenarios

A file system is a container that store files in SFS. You need to create a file system before storing data in SFS.

The following describes how to call the API for [Creating a File System](#) in a region. For details, see [Calling General Purpose File System APIs](#).

Prerequisites

- You have obtained the AK and SK. For details, see [Obtaining Access Keys \(AK/SK\)](#).
- You have planned the region where you want to create a file system and obtained the endpoint required for API calls. For details, see [Regions and Endpoints](#).

Once a region is determined, it cannot be modified after the bucket is created.

Creating a File System Named filesystem001 in the a1 Region

In this example, an Apache HttpClient is used.

```
package com.sfsclient;

import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.List;
import java.util.Map;

public class TestMain {
    //Obtain an AK/SK pair using environment variables or import the AK/SK pair in other ways. Using hard
    coding may result in leakage.
    //Obtain an AK/SK pair on the management console.
    public static String accessKey = System.getenv("HUAWEICLOUD_SDK_AK");
    public static String securityKey = System.getenv("HUAWEICLOUD_SDK_SK");
```

```
public static String region = "cn-east-3"; // The region where you plan to create the file system.

public static String endpoint = "sfs3.a1.region.com"; // The access address of General Purpose File System.

public static String createSfsBody =
    "<CreateBucketConfiguration >\n" +
    "  <Location>" + region + "</Location>\n" +
    "</CreateBucketConfiguration>";

public static void main(String[] str) {

    createFileSystem();

}

private static void createFileSystem() {
    // The file system name.
    String fileName = "example-sfs-001";

    String httpMethod = "PUT";
    String date = DateUtils.formatDate(System.currentTimeMillis());
    String contentType = "application/xml";

    /**Calculate the signature based on the request.**/
    String contentMD5 = "";
    String canonicalizedHeaders = "x-obs-bucket-type:SFS";
    String canonicalizedResource = "/" + fileName ;

    // Content-MD5 and Content-Type fields do not contain line breaks. The data format is RFC 1123,
    which is the same as the time in the request.
    String stringToSign = httpMethod + "\n" +
        contentMD5 + "\n" +
        contentType + "\n" +
        date + "\n" +
        canonicalizedHeaders + "\n" + canonicalizedResource;

    System.out.printf("StringToSign:\n[%s]\n\n", stringToSign);

    HttpURLConnection conn = null;

    try {
        String signature = Signature.signWithHmacSha1 (securityKey, stringToSign);
        String authorization= "OBS " + accessKey + ":" + signature;
        System.out.printf("authorization:%s\n\n", authorization);

        URL url = new URL("http://" + endpoint + "/" + fileName);
        conn = (HttpURLConnection) url.openConnection();

        // Add a signature header.
        conn.setRequestMethod(httpMethod);
        conn.setRequestProperty("Date", date);
        conn.setRequestProperty("Content-Type", contentType);
        conn.setRequestProperty("x-obs-bucket-type", "SFS");
        conn.setRequestProperty("Authorization", authorization);
        conn.setDoOutput(true);

        // Add a body.
        OutputStream out = conn.getOutputStream();
        out.write(createSfsBody.getBytes());
        out.flush();
        out.close();

        String status = conn.getHeaderField(null);
        System.out.println(status);

        // Output the response message.
        Map<String, List<String>> headers = conn.getHeaderFields();
        for (Map.Entry<String, List<String>> entry : headers.entrySet()) {
            String key = entry.getKey();
            List<String> values = entry.getValue();
```



```
        if (key != null) {
            for (String value : values) {
                System.out.println(key + ": " + value);
            }
        }
    }
    // Handle the request error.
    int statusCode = conn.getResponseCode();
    if (statusCode != HttpURLConnection.HTTP_OK && statusCode !=
HttpURLConnection.HTTP_NO_CONTENT) {
        InputStream errorStream = conn.getErrorStream();
        BufferedReader reader = new BufferedReader(new InputStreamReader(errorStream));
        StringBuilder responseBody = new StringBuilder();
        String line;
        while ((line = reader.readLine()) != null) {
            responseBody.append(line);
        }
        reader.close();

        System.out.println("Error: " + responseBody);
    }
} catch (IOException e) {
    e.printStackTrace();
} finally {
    if (conn != null){
        conn.disconnect();
    }
}
}
```

The format of the **Date** header field **DateUtils** is as follows:

```
package com.sfsclient;

import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Locale;
import java.util.TimeZone;

public class DateUtils {

    public static String formatDate(long time)
    {
        DateFormat serverDateFormat = new SimpleDateFormat("EEE, dd MMM yyyy HH:mm:ss z",
Locale.ENGLISH);
        serverDateFormat.setTimeZone(TimeZone.getTimeZone("GMT"));
        return serverDateFormat.format(time);
    }
}
```

The method of calculating the signature character string is as follows:

```
package com.sfsclient;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.io.UnsupportedEncodingException;
import java.security.NoSuchAlgorithmException;
import java.security.InvalidKeyException;
import java.util.Base64;

public class Signature {
    public static String signWithHmacSha1(String sk, String canonicalString) throws
UnsupportedEncodingException {

        try {
            SecretKeySpec signingKey = new SecretKeySpec(sk.getBytes("UTF-8"), "HmacSHA1");
```

```
Mac mac = Mac.getInstance("HmacSHA1");
mac.init(signingKey);
return Base64.getEncoder().encodeToString(mac.doFinal(canonicalString.getBytes("UTF-8")));
} catch (NoSuchAlgorithmException | InvalidKeyException | UnsupportedEncodingException e) {
    e.printStackTrace();
}
return null;
}
```

7.2 Listing File Systems

Scenarios

If you want to view information about all file systems created by yourself, you can call the API for listing file systems.

The following describes how to call the API for [Listing File Systems](#). For details, see [Getting Started with General Purpose File System](#).

Prerequisites

- You have obtained the AK and SK. For details, see [Obtaining Access Keys \(AK/SK\)](#).
- You have specified the region where you want to list file systems and obtained the endpoint required for API calls. For details, see [Regions and Endpoints](#).

Listing File Systems in the a1 Region

In this example, an Apache HttpClient is used.

```
package com.sfsclient;

import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.List;
import java.util.Map;

public class TestMain {
    //Obtain an AK/SK pair using environment variables or import the AK/SK pair in other ways. Using hard
    coding may result in leakage.
    //Obtain an AK/SK pair on the management console.
    public static String accessKey = "ACCESS_KEY_ID";
    public static String securityKey = "SECRET_ACCESS_KEY_ID";
    public static String endpoint = "sfs3.a1.region.com"; // The access address of General Purpose File System.

    public static void main(String[] str) {
        createFileSystem();
    }
    private static void listFileSystem() {
        String httpMethod = "GET";
        String date = DateUtils.formatDate(System.currentTimeMillis());

        /**Calculate the signature based on the request.**/
        String contentMD5 = "";
        String contentType = "";
        String canonicalizedHeaders = "x-obs-bucket-type:SFS";
        String canonicalizedResource = "" ;
    }
}
```

```
// Content-MD5 and Content-Type fields do not contain line breaks. The data format is RFC 1123,
which is the same as the time in the request.
String stringToSign = httpMethod + "\n" +
    contentMD5 + "\n" +
    contentType + "\n" +
    date + "\n" +
    canonicalizedHeaders + "\n" + canonicalizedResource;

System.out.printf("StringToSign:\n[%s]\n\n", stringToSign);

URLConnection conn = null;

try {
    String signature = Signature.signWithHmacSha1 (securityKey, stringToSign);
    String authorization= "OBS " + accessKey + ":" + signature;
    System.out.printf("authorization:%s\n\n", authorization);

    // Create an HTTP request.
    URL url = new URL("http://" + endpoint);
    conn = (URLConnection) url.openConnection();

    // Add a signature header.
    conn.setRequestMethod(httpMethod);
    conn.setRequestProperty("Date", date);
    conn.setRequestProperty("Content-Type", contentType);
    conn.setRequestProperty("x-obs-bucket-type", "SFS");
    conn.setRequestProperty("Authorization", authorization);
    conn.setDoOutput(true);

    String status = conn.getHeaderField(null);
    System.out.println(status);

    // Output the response message.
    Map<String, List<String>> headers = conn.getHeaderFields();
    for (Map.Entry<String, List<String>> entry : headers.entrySet()) {
        String key = entry.getKey();
        List<String> values = entry.getValue();
        if (key != null) {
            for (String value : values) {
                System.out.println(key + ": " + value);
            }
        }
    }
    // Process the returned content.
    int statusCode = conn.getResponseCode();
    if (statusCode == HttpURLConnection.HTTP_OK) {
        InputStream responseStream = conn.getInputStream();
        BufferedReader reader = new BufferedReader(new InputStreamReader(responseStream));

        StringBuilder responseBody = new StringBuilder();
        String line;
        while ((line = reader.readLine()) != null) {
            responseBody.append(line);
        }
        reader.close();

        System.out.println("responseBody: " + responseBody);
    } else {
        System.out.println("Error: " + statusCode);
    }
} catch (IOException e) {
    e.printStackTrace();
} finally {
    if (conn != null){
        conn.disconnect();
    }
}
}
```

The format of the **Date** header field **DateUtils** is as follows:

```
package com.sfsclient;

import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Locale;
import java.util.TimeZone;

public class DateUtils {

    public static String formatDate(long time)
    {
        DateFormat serverDateFormat = new SimpleDateFormat("EEE, dd MMM yyyy HH:mm:ss z",
Locale.ENGLISH);
        serverDateFormat.setTimeZone(TimeZone.getTimeZone("GMT"));
        return serverDateFormat.format(time);
    }
}
```

The method of calculating the signature character string is as follows:

```
package com.sfsclient;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.io.UnsupportedEncodingException;
import java.security.NoSuchAlgorithmException;
import java.security.InvalidKeyException;
import java.util.Base64;

public class Signature {

    public static String signWithHmacSha1(String sk, String canonicalString) throws
UnsupportedEncodingException {

        try {
            SecretKeySpec signingKey = new SecretKeySpec(sk.getBytes("UTF-8"), "HmacSHA1");
            Mac mac = Mac.getInstance("HmacSHA1");
            mac.init(signingKey);
            return Base64.getEncoder().encodeToString(mac.doFinal(canonicalString.getBytes("UTF-8")));
        } catch (NoSuchAlgorithmException | InvalidKeyException | UnsupportedEncodingException e) {
            e.printStackTrace();
        }
        return null;
    }
}
```

8 SFS Capacity-Oriented APIs

8.1 API Version Queries

8.1.1 Querying All API Versions

Function

This API is used to query all available versions of APIs provided by SFS.

To support function extension, SFS APIs can be distinguished by version. SFS has two types API version IDs:

Major version: Independent URL. For example: **v1** and **v2**.

Microversion: with the HTTP request header **X-Openstack-Manila-API-Version: *Microversion ID***. For example: **X-Openstack-Manila-API-Version: 2.4**.

NOTE

This API does not require authentication.

URI

- GET /
- Parameter description

None

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
- None

- Example request
GET https://{endpoint}/

Response

- Parameter description

Parameter	Type	Description
versions	Array of objects	Lists objects of all available API versions, including v1 and v2 .

- Description of the **version** field

Parameter	Type	Description
id	String	Specifies the common name of the version.
updated	String	Specifies the UTC time when the API is last modified. The format is YYYY-MM-DDTHH:MM:SSZ.
status	String	Specifies the API version status, including: <ul style="list-style-type: none"> • CURRENT: indicates that the current API is the preferred version. • SUPPORTED: indicates that the current version is an earlier version which is still supported. • DEPRECATED: indicates that the current version is a deprecated version that may be deleted later.
links	Array of objects	Specifies the links of shared file systems. For details, see the description of the links field.
media-types	Array of objects	Specifies the media types supported by the API. For details, see the description of the media-types field.
version	String	If the API in the current version supports microversions, this parameter is the latest microversion. If microversions are not supported, this parameter is an empty string.
min_version	String	If the API in the current version supports microversions, this parameter is the earliest microversion. If microversions are not supported, this parameter is an empty string.

- Description of the **links** field

Parameter	Type	Description
href	String	Specifies the API access path, which is used as a reference.
type	String	Specifies the type of the text returned by the reference API.
rel	String	Specifies the additional description on links.

- Description of the **media-types** field

Parameter	Type	Description
base	String	Specifies the basic text type.
type	String	Specifies the text type.

- Example response

```
{
  "versions": [
    {
      "status": "CURRENT",
      "updated": "2015-08-27T11:33:21Z",
      "links": [
        {
          "href": "http://docs.openstack.org/",
          "type": "text/html",
          "rel": "describedby"
        },
        {
          "href": "https://sfs.region.www.t-systems.com/v2/",
          "rel": "self"
        }
      ],
      "min_version": "2.0",
      "version": "2.28",
      "media-types": [
        {
          "base": "application/json",
          "type": "application/vnd.openstack.share+json;version=1"
        }
      ],
      "id": "v2.0"
    }
  ]
}
```

Status Codes

- Normal
300
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.

Status Code	Description
400 Bad Request	Invalid input: The post-deduction capacity must be larger than 0 and smaller than the current capacity. (Current capacity: <i>XX</i> ; post-deduction capacity: <i>XX</i>)
400 Bad Request	Invalid input: The post-expansion capacity must be larger than the current capacity. (Current capacity: <i>XX</i> ; post-expansion capacity: <i>XX</i>)
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	The request is not completed because of a service error.
501 Not Implemented	The request is not completed because the server does not support the requested function.
502 Bad Gateway	The request is not completed because the request is invalid.
503 Service Unavailable	The request is not completed because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.1.2 Querying Details About an API Version

Function

This API is used for querying details about an API version.

URI

- GET `/api_version/`
- Parameter description

Parameter	Mandatory	Type	Description
api_version	Yes	String	Specifies the API version, which can be v1 or v2 .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
GET `https://{endpoint}/v2/`

Response

- Parameter description

Parameter	Type	Description
versions	Object	List objects of all available API versions

- Description of the **version** field

Parameter	Type	Description
id	String	Specifies the common name of the version.
updated	String	Specifies the UTC time when the API is last modified. The format is YYYY-MM-DDTHH:MM:SSZ.

Parameter	Type	Description
status	String	Specifies the API version status, including: <ul style="list-style-type: none"> • CURRENT: indicates that the current API is the preferred version. • SUPPORTED: indicates that the current version is an earlier version which is still supported. • DEPRECATED: indicates that the current version is a deprecated version that may be deleted later.
links	Array of objects	Specifies the links of shared file systems. For details, see the description of the links field.
media-types	Array of objects	Specifies the media types supported by the API. For details, see the description of the media-types field.
version	String	If the API in the current version supports microversions, this parameter is the latest microversion. If microversions are not supported, this parameter is an empty string.
min_version	String	If the API in the current version supports microversions, this parameter is the earliest microversion. If microversions are not supported, this parameter is an empty string.

– Description of the **links** field

Parameter	Type	Description
href	String	Specifies the API access path, which is used as a reference.
type	String	Specifies the type of the text returned by the reference API.
rel	String	Specifies the additional description on links.

– Description of the **media-types** field

Parameter	Type	Description
base	String	Specifies the basic text type.
type	String	Specifies the text type.

- Example response

```
{
  "versions": [
    {
      "status": "CURRENT",
      "updated": "2015-08-27T11:33:21Z",
      "links": [
        {
          "href": "http://docs.openstack.org/",
          "type": "text/html",
          "rel": "describedby"
        },
        {
          "href": "https://sfs.region.www.t-systems.com/v2/",
          "rel": "self"
        }
      ],
      "min_version": "2.0",
      "version": "2.28",
      "media-types": [
        {
          "base": "application/json",
          "type": "application/vnd.openstack.share+json;version=1"
        }
      ],
      "id": "v2.0"
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
400 Bad Request	Invalid input: The post-deduction capacity must be larger than 0 and smaller than the current capacity. (Current capacity: <i>XX</i> ; post-deduction capacity: <i>XX</i>)
400 Bad Request	Invalid input: The post-expansion capacity must be larger than the current capacity. (Current capacity: <i>XX</i> ; post-expansion capacity: <i>XX</i>)
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.

Status Code	Description
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	The request is not completed because of a service error.
501 Not Implemented	The request is not completed because the server does not support the requested function.
502 Bad Gateway	The request is not completed because the request is invalid.
503 Service Unavailable	The request is not completed because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2 File Systems

8.2.1 Creating a Shared File System

Function

This API is used to create a shared file system. After the file system is created, you need to mount the file system to ECSs to achieve shared file storage. For details about mounting the file system, see [SFS Getting Started](#).

NOTE

This API is an asynchronous API. If the returned status code is **200**, the API request is successfully delivered and received. Later, you can query the status and path of the shared file system by referring to [Querying Details About a Shared File System](#) to identify whether the creation is complete and successful. If the status of the shared file system becomes **available** or the shared path is generated, the creation is successful.

NOTICE

After a shared file system is created successfully, it can be used only after you add share access rules by referring to [Adding a File System Access Rule](#).

URI

- POST /v2/{project_id}/shares
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
share	Yes	Object	For details, see the description of the share field.

- Description of the **share** field

Parameter	Mandatory	Type	Description
share_proto	Yes	String	Specifies the file sharing protocol. The value can be NFS (for Linux OS).
size	Yes	Integer	Specifies the size (GB) of the shared file system. The applied capacity of the shared file system cannot be larger than the allowed quota. To view the allowed quota, see Quota Management .
name	No	String	Specifies the name of the shared file system, which contains 0 to 255 characters and can contain only letters, digits, hyphens (-), and underscores (_).
description	No	String	Specifies the description of the shared file system, which contains 0 to 255 characters and can contain only letters, digits, hyphens (-), and underscores (_).

Parameter	Mandatory	Type	Description
is_public	No	Boolean	(Supported by API versions from v2.8 to v2.42). Specifies whether a file system can be publicly seen. If it is set to true , the file system can be seen publicly. If it is set to false , the file system can be seen privately. The default value is false .
availability_zone	No	String	Specifies the availability zone name. If this parameter is left blank, the default availability zone will be used. If the default availability zone contains no storage resources, the creation of the shared file system fails. The value contains 0 to 255 characters.
metadata	No	Object	<p>Specifies the metadata information used to create the shared file system. The value consists of one or more key and value pairs organized as a dictionary of strings. For details, see the description of the metadata field.</p> <p>CAUTION</p> <ul style="list-style-type: none"> For security concerns, the API for modifying the metadata field is not opened yet. Therefore, ensure that the parameters and values are correct when creating a shared file system with data encryption using the metadata field. Unless otherwise specified (for example, #sfs_crypt_key_id), the keys that comply with the following rules in the metadata field are for internal use of the system. Do not customize the settings to avoid internal system errors caused by conflicts with system predefined keys. <ul style="list-style-type: none"> Key share_used Keys that start with #sfs

- Description of **metadata** fields (creating a shared file system with the encryption function)

When creating a shared file system with the encryption function, obtain the key ID, domain ID, and key alias of the encryption key using the HTTPS

request by referring to section **Querying the List of CMKs** in the *Data Encryption Workshop API Reference*. Then, in the **metadata** field, set the key-value pairs according to the following table. Ensure that the key-value pairs in the **metadata** field are correct.

To create a shared file system with the encryption function, all parameters in the following table are mandatory. If the encryption function is not needed, these parameters are optional.

Key	Value Type	Mandatory	Description
#sfs_crypt_key_id	String	Yes	Specifies the encryption key ID. If this field, #sfs_crypt_domain_id , and #sfs_crypt_alias exist at the same time, the data encryption function is enabled.
#sfs_crypt_domain_id	String	Yes	Specifies the tenant domain ID. If this field, #sfs_crypt_key_id , and #sfs_crypt_alias exist at the same time, the data encryption function is enabled.
#sfs_crypt_alias	String	Yes	Specifies the encryption key alias. If this field, #sfs_crypt_key_id , and #sfs_crypt_domain_id exist at the same time, the data encryption function is enabled.

 **NOTE**

- You are advised to use the default key **sfs/default** to create an encrypted shared file system. For details, see section "File System Encryption" and "Encryption" in the *Scalable File Service User Guide*.
- When you create a file system with an enterprise project ID, the following key is required in the metadata.

Key	Value Type	Mandatory	Description
enterprise_project_id	String	Yes	Specifies the enterprise project ID. When the Enterprise Project function is enabled, enterprise_project_id is used as a key for specifying the enterprise project ID of the shared file system.

- Example request: POST https://{endpoint}/v2/16e1ab15c35a457e9c2b2aa189f544e1/shares

Creating a 1-GB NFS file system that can only be seen privately

```
{
  "share": {
    "name": "test",
    "description": "test description",
    "share_proto": "NFS",
    "share_network_id": null,
    "size": 1,
    "is_public": false
  }
}
```

- Example request (creating a shared file system with data encryption function): POST https://{endpoint}/v2/16e1ab15c35a457e9c2b2aa189f544e1/shares

Creating a 1-GB, encrypted NFS file system that can only be seen privately

```
{
  "share": {
    "name": "test",
    "description": "test description",
    "metadata": {
      "#sfs_crypt_key_id": "9130c90d-73b8-4203-b790-d49f98d503df",
      "#sfs_crypt_domain_id": "3b2d9670690444c582942801ed7d457b",
      "#sfs_crypt_alias": "sfs/default"
    },
    "share_proto": "NFS",
    "share_network_id": null,
    "size": 1,
    "is_public": false
  }
}
```

- Example request (with Enterprise Project enabled): POST https://{endpoint}/v2/16e1ab15c35a457e9c2b2aa189f544e1/shares

Creating a 1-GB NFS file system that can only be seen privately and adding it to an enterprise project

```
{
  "share": {
    "share_type": null,
    "name": "test",
    "snapshot_id": null,
    "description": "test description",
    "metadata": {
      "enterprise_project_id": "9130c90d-73b8-4203-b790-d49f98d503df"
    },
    "share_proto": "NFS",
    "share_network_id": null,
    "size": 1,
    "is_public": false
  }
}
```

Response

- Parameter description

Parameter	Type	Description
share	Object	For details, see the description of the share field.

- Description of the **share** field

Parameter	Type	Description
links	Array	Specifies the links of shared file systems.
availability_zone	String	Specifies the availability zone.
share_server_id	String	Specifies the ID for managing share services.
id	String	Specifies the ID of the shared file system.
size	Integer	Specifies the size (GB) of the shared file system.
project_id	String	Specifies the ID of the project to which the shared file system belongs.
metadata	Object	Sets one or more metadata key and value pairs as a dictionary of strings. The value of the share_used key indicates the file system used capacity, in bytes. The value of the enterprise_project_id key indicates the ID of the enterprise project that the file system belongs to.
status	String	Specifies the status of the shared file system.
description	String	Describes the shared file system.
host	String	Specifies the name of the host.
name	String	Specifies the name of the shared file system.
created_at	String	Specifies the date and time stamp when the shared file system was created.
share_proto	String	Specifies the protocol for sharing file systems.
share_type_name	String	Specifies the storage service type assigned for the shared file system, such as high-performance storage (composed of SSDs) and large-capacity storage (composed of SATA disks). This field is supported by API v2.6 and later versions.
share_type	String	Specifies the ID of the file system type.
volume_type	String	Specifies the volume type. The definition of this parameter is the same as that of share_type .

Parameter	Type	Description
export_locations	Array	Lists the mount locations. Currently, only a single mount location is supported. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .
export_location	String	Specifies the mount location. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .
is_public	Boolean	Specifies the visibility level of the shared file system. If true is returned, the file system can be seen publicly. If false is returned, the file system can be seen privately. The default value is false .

- Example response

```
{
  "share": {
    "status": "creating",
    "project_id": "16e1ab15c35a457e9c2b2aa189f544e1",
    "name": "share_London",
    "share_type": "25747776-08e5-494f-ab40-a64b9d20d8f7",
    "availability_zone": "az1.dc1",
    "created_at": "2015-09-18T10:25:24.533287",
    "export_location": null,
    "links": [
      {
        "href": "http://192.168.198.54:8786/v2/16e1ab15c35a457e9c2b2aa189f544e1/shares/011d21e2-fbc3-4e4a-9993-9ea223f73264",
        "rel": "self"
      },
      {
        "href": "http://192.168.198.54:8786/16e1ab15c35a457e9c2b2aa189f544e1/shares/011d21e2-fbc3-4e4a-9993-9ea223f73264",
        "rel": "bookmark"
      }
    ],
    "share_network_id": null,
    "export_locations": [],
    "share_proto": "NFS",
    "host": null,
    "volume_type": "default",
    "snapshot_id": null,
    "is_public": true,
    "metadata": {
      "project": "my_app",
      "aim": "doc"
    },
    "id": "011d21e2-fbc3-4e4a-9993-9ea223f73264",
    "size": 1,
    "description": "My custom share London"
  }
}
```

 NOTE

When the client receives the system response, the shared file system is still being created. For this reason, the shared path cannot be queried immediately. You can use the API of [Querying Mount Locations of a Shared File System](#) to query the shared path after the creation is complete.

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
413 Quota Exceeded	Insufficient user quota.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2.2 Querying All Shared File Systems

Function

This API is used to list the basic information of all shared file systems.

URI

- GET /v2/{project_id}/shares?
all_tenants={all_tenants}&status={status}&limit={limit}&offset={offset}&sort_key={sort_key}&sort_dir={sort_dir}&project_id={project_id}&is_public={is_public}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
all_tenants	No (query parameter)	Boolean	This parameter is available only to users with administrator permissions. Specifies whether to list shared file systems of all tenants. To list shared file systems of all tenants, set it to 1 . To list shared file systems only of the current tenant, set it to 0 .
project_id	No (query parameter)	String	This parameter is available only to users with administrator permissions. Specifies the ID of the project to which the shared file system belongs. This parameter needs to be used together with all_tenants .

Parameter	Mandatory	Type	Description
status	No (query parameter)	String	<p>Filters shared file systems by status. Possible values are:</p> <ul style="list-style-type: none"> • creating: The shared file system is being created. • error: The shared file system fails to be created. • available: The shared file system is available. • deleting: The shared file system is being deleted. • error_deleting: The shared file system fails to be deleted. • extending: The shared file system is being expanded. • extending_error: The shared file system fails to be expanded. • shrinking: The shared file system is being shrunk. • shrinking_error: The shared file system fails to be shrunk. • shrinking_possible_data_loss_error: The shared file system fails to be shrunk due to data loss. • manage_starting: Shared file system management starts. • manage_error: The shared file system fails to be managed. • unmanage_starting: Canceling shared file system management starts. • unmanage_error: Failed to cancel shared file system management. • unmanaged: The shared file system is not managed.
limit	No (query parameter)	Integer	<p>Specifies the maximum number of shared file systems that can be returned. If this parameter is not specified, all the shared file systems are returned by default.</p>

Parameter	Mandatory	Type	Description
offset	No (query parameter)	Integer	Specifies the offset to define the start point of shared file system listing. The value must be greater than or equal to 0 .
sort_key	No (query parameter)	String	Specifies the keyword for sorting the queried shared file systems. Possible values are id , status , size , host , share_proto , availability_zone_id , user_id , project_id , created_at , updated_at , display_name , name , share_type_id , share_network_id , and snapshot_id . By default, the value is sorted by created_at .
sort_dir	No (query parameter)	String	Specifies the direction to sort shared file systems. Possible values are asc (ascending) and desc (descending).
is_public	No (query parameter)	String	When this parameter is set to true , the current tenant can query all its own shared file systems and other tenants' shared file systems whose is_public is set to true . When this parameter is set to false , the current tenant can query only the shared file systems owned by the tenant.
enterprise_project_id	No (query parameter)	String	Specifies the enterprise project to which the shared file systems are bound. To query the current user's shared file systems binding to all enterprise projects, use the all_granted_eps parameter.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None

- Example request
GET https://{endpoint}/v2/16e1ab15c35a457e9c2b2aa189f544e1/shares

Response

- Parameter description

Parameter	Type	Description
shares	Array of objects	For details, see the description of the share field.

- Description of the **share** field

Parameter	Type	Description
id	String	Specifies the ID of the shared file system.
links	Array of objects	Specifies the request link information of the shared file system.
name	String	Specifies the name of the shared file system.

- Example response

```
{
  "shares": [
    {
      "id": "1390cb29-539b-4926-8953-d8d6b106071a",
      "links": [
        {
          "href": "https://192.168.196.47:8796/v2/f24555bfcf3146ca936d21bcb548687e/shares/1390cb29-539b-4926-8953-d8d6b106071a",
          "rel": "self"
        },
        {
          "href": "https://192.168.196.47:8796/f24555bfcf3146ca936d21bcb548687e/shares/1390cb29-539b-4926-8953-d8d6b106071a",
          "rel": "bookmark"
        }
      ],
      "name": null
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.

Status Code	Description
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2.3 Querying Details About All Shared File Systems

Function

This API is used to query the details about all shared file systems.

URI

- GET /v2/{project_id}/shares/detail?
all_tenants={all_tenants}&project_id={project_id}&status={status}&limit={limit}
&offset={offset}&sort_key={sort_key}&sort_dir={sort_dir}&is_public={is_public}
&name={name}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
all_tenants	No (query parameter)	Integer	(Administrators only) Specifies whether to list shared file systems of all tenants. To list shared file systems of all tenants, set it to 1 . To list shared file systems only of the current tenant, set it to 0 .
project_id	No (query parameter)	String	Specifies the ID of the tenant who creates the shared file system. This parameter is used together with all_tenants .
status	No (query parameter)	String	Filters shared file systems by status. Possible values are creating, error, available, deleting, error_deleting, manage_starting, manage_error, unmanage_starting, unmanage_error, unmanaged, extending, extending_error, shrinking, shrinking_error, and shrinking_possible_data_loss_error .
limit	No (query parameter)	Integer	Specifies the maximum number of shared file systems that can be returned.
offset	No (query parameter)	Integer	Specifies the offset to define the start point of shared file system listing.
sort_key	No (query parameter)	String	Specifies the keyword for sorting the queried shared file systems. Possible values are id, status, size, host, share_proto, availability_zone_id, user_id, project_id, created_at, updated_at, display_name, name, share_type_id, share_network_id, and snapshot_id .

Parameter	Mandatory	Type	Description
sort_dir	No (query parameter)	String	Specifies the direction to sort shared file systems. Possible values are asc (ascending) and desc (descending).
is_public	No (query parameter)	String	When this parameter is set to true , the current tenant can query all its own shared file systems and other tenants' shared file systems whose is_public is set to true . When this parameter is set to false , the current tenant can query only the shared file systems owned by the tenant.
enterprise_project_id	No (query parameter)	String	Specifies the enterprise project to which the shared file systems are bound. To query the current user's shared file systems binding to all enterprise projects, use the all_granted_eps parameter.
name	No (query parameter)	String	Specifies the field used for fuzzy filtering based on the name of a shared file system. This field is supported by API v2.36 and later versions.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
shares	Array of objects	Specifies the list of the share objects.

- Description of the **share** field

Parameter	Type	Description
links	Array	Specifies the links of shared file systems.
availability_zone	String	Specifies the availability zone.
share_server_id	String	Specifies the ID for managing share services.
share_network_id	String	Specifies the ID of the share network. This parameter is reserved, because share network management is not supported currently.
snapshot_id	String	Specifies the ID of the source snapshot that is used to create the shared file system. This parameter is reserved, because snapshots are not supported currently.
snapshot_support	Boolean	Specifies whether snapshots are supported. This parameter is reserved, because snapshots are not supported currently. This field is supported by API v2.2 and later versions.
id	String	Specifies the ID of the shared file system.
size	Integer	Specifies the size (GB) of the shared file system.
consistency_group_id	String	Specifies the ID of the consistency group. This parameter is reserved, because consistency groups are not supported currently. This field is supported by API versions from v2.31 to v2.42.
project_id	String	Specifies the ID of the project to which the shared file system belongs.
metadata	Object	Sets one or more metadata key and value pairs as a dictionary of strings. The value of the share_used key indicates the file system used capacity, in bytes. The used capacity will not be displayed if less than 1 MB of the SFS Capacity-Oriented file system is used. The value of the enterprise_project_id key indicates the ID of the enterprise project that the file system belongs to.
status	String	Specifies the status of the shared file system.

Parameter	Type	Description
task_state	String	Specifies the data migration status. This parameter is reserved, because data migration is not supported currently. This field is supported by API v2.5 and later versions.
has_replicas	Boolean	Specifies whether replicas exist. This parameter is reserved, because replication is not supported currently. This field is supported by API versions from v2.11 to v2.42.
replication_type	String	Specifies the replication type. This parameter is reserved, because replication is not supported currently. This field is supported by API versions from v2.11 to v2.42.
description	String	Describes the shared file system.
host	String	Specifies the name of the host.
name	String	Specifies the name of the shared file system.
created_at	String	Specifies the date and time stamp when the shared file system was created.
share_proto	String	Specifies the protocol for sharing file systems.
share_type_name	String	Specifies the storage service type assigned for the shared file system, such as high-performance storage (composed of SSDs) and large-capacity storage (composed of SATA disks). This field is supported by API v2.6 and later versions.
share_type	String	Specifies the ID of the file system type.
volume_type	String	Specifies the volume type. The definition of this parameter is the same as that of share_type .
export_locations	Array	Lists the mount locations. Currently, only a single mount location is supported. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .
export_location	String	Specifies the mount location. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .

Parameter	Type	Description
is_public	Boolean	Specifies the visibility level of the shared file system. If true is returned, the file system can be seen publicly. If false is returned, the file system can be seen privately. The default value is false .

- Example response

```
{
  "shares": [
    {
      "links": [
        {
          "href": "https://192.168.170.97:8796/v2/61b01a94b84448cfac2424e46553d7e7/shares/54d0bac6-45c8-471c-bf0d-16ffd81ef88a",
          "rel": "self"
        },
        {
          "href": "https://192.168.170.97:8796/61b01a94b84448cfac2424e46553d7e7/shares/54d0bac6-45c8-471c-bf0d-16ffd81ef88a",
          "rel": "bookmark"
        }
      ],
      "export_location": "sfs.dong.com:/share-e1c2d35e",
      "availability_zone": "az1.dc1",
      "share_network_id": null,
      "snapshot_id": null,
      "id": "54d0bac6-45c8-471c-bf0d-16ffd81ef88a",
      "size": 1,
      "share_type": "default",
      "": null,
      "project_id": "da0f615c35eb4d72812d1547a77b5394",
      "metadata": {
        "share_used": "1048576000000",
      },
      "status": "available",
      "description": "test description",
      "export_locations": ["sfs.dong.com:/share-e1c2d35e"],
      "host": "DJ01@9656beb1-7ce2-4c46-9911-eed51ab632bf#9656beb1-7ce2-4c46-9911-eed51ab632bf",
      "is_public": false,
      "name": "cl01",
      "created_at": "2017-07-07T03:15:06.858662",
      "share_proto": "NFS",
      "volume_type": "default"
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.

Status Code	Description
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2.4 Querying Details About a Shared File System

Function

This API is used to query the details about a shared file system.

URI

- GET /v2/{project_id}/shares/{share_id}
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
share	Object	Specifies the share objects.

- Description of the **share** field

Parameter	Type	Description
links	Array	Specifies the links of shared file systems.
availability_zone	String	Specifies the availability zone.
share_server_id	String	Specifies the ID for managing share services.
share_network_id	String	Specifies the ID of the share network. This parameter is reserved, because share network management is not supported currently.
snapshot_id	String	Specifies the ID of the source snapshot that is used to create the shared file system. This parameter is reserved, because snapshots are not supported currently.

Parameter	Type	Description
snapshot_support	Boolean	Specifies whether snapshots are supported. This parameter is reserved, because snapshots are not supported currently. This field is supported by API v2.2 and later versions.
id	String	Specifies the ID of the shared file system.
size	Integer	Specifies the size (GB) of the shared file system.
consistency_group_id	String	Specifies the ID of the consistency group. This parameter is reserved, because consistency groups are not supported currently. This field is supported by API versions from v2.31 to v2.42.
project_id	String	Specifies the ID of the project to which the shared file system belongs.
metadata	Object	Sets one or more metadata key and value pairs as a dictionary of strings. The value of the share_used key indicates the file system used capacity, in bytes. The used capacity will not be displayed if less than 1 MB of the SFS Capacity-Oriented file system is used. The value of the enterprise_project_id key indicates the ID of the enterprise project that the file system belongs to.
status	String	Specifies the status of the shared file system.
task_state	String	Specifies the data migration status. This parameter is reserved, because data migration is not supported currently. This field is supported by API v2.5 and later versions.
has_replicas	Boolean	Specifies whether replicas exist. This parameter is reserved, because replication is not supported currently. This field is supported by API versions from v2.11 to v2.42.
replication_type	String	Specifies the replication type. This parameter is reserved, because replication is not supported currently. This field is supported by API versions from v2.11 to v2.42.
description	String	Describes the shared file system.

Parameter	Type	Description
host	String	Specifies the name of the host.
name	String	Specifies the name of the shared file system.
created_at	String	Specifies the date and time stamp when the shared file system was created.
share_proto	String	Specifies the protocol for sharing file systems.
share_type_name	String	Specifies the storage service type assigned for the shared file system, such as high-performance storage (composed of SSDs) and large-capacity storage (composed of SATA disks). This field is supported by API v2.6 and later versions.
share_type	String	Specifies the ID of the file system type.
volume_type	String	Specifies the volume type. The definition of this parameter is the same as that of share_type .
export_locations	Array	Lists the mount locations. Currently, only a single mount location is supported. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .
export_location	String	Specifies the mount location. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.9 .
is_public	Boolean	Specifies the visibility level of the shared file system. If true is returned, the file system can be seen publicly. If false is returned, the file system can be seen privately. The default value is false .

- Example response

```
{
  "share": {
    "status": "available",
    "share_type_name": "sla",
    "description": "My custom share London",
    "links": [
      {
        "href": "https://192.168.196.47:8796/v2/07412155bf474db9a2f697fd978593d7/shares/f26d867f-9876-433d-8db2-25d210f29309",
        "rel": "self"
      },
      {
        "href": "https://192.168.196.47:8796/07412155bf474db9a2f697fd978593d7/shares/f26d867f-9876-433d-8db2-25d210f29309",

```

```

    "rel": "bookmark"
  }
],
"availability_zone": "az1.dc1",
"share_network_id": null,
"share_server_id": null,
"share_group_id": null,
"host": "DJ38@a4588256-3880-4136-b3c9-4c3aade8a84b#a4588256-3880-4136-
b3c9-4c3aade8a84b",
"revert_to_snapshot_support": null,
"access_rules_status": "active",
"snapshot_id": null,
"create_share_from_snapshot_support": null,
"is_public": false,
"task_state": null,
"snapshot_support": true,
"id": "f26d867f-9876-433d-8db2-25d210f29309",
"size": 1,
"source_share_group_snapshot_member_id": null,
"user_id": "daa3f8f8d7254465841da769298a76f6",
"name": "luzhongguo_1",
"share_type": "8ae4e74e-83f4-4980-8ab8-e637f9294e0b",
"has_replicas": false,
"replication_type": null,
"created_at": "2018-12-25T08:45:22.525899",
"share_proto": "NFS",
"volume_type": "sla",
"mount_snapshot_support": null,
"project_id": "07412155bf474db9a2f697fd978593d7",
"metadata": {
  "share_key": "test",
  "share_used": "1",
}
}
}

```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.

Status Code	Description
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2.5 Querying Mount Locations of a Shared File System

Function

This API is used to query mount locations of a shared file system.

NOTE

This API exists only when **X-Openstack-Manila-API-Version** in the request header is greater than or equal to 2.9. The following is an example request sent by the **curl** command: `curl -k -i -X GET https://192.168.196.47:8786/v2/13c7ff9a479c4e3599f4331d9e4a1835/shares/2a8c5470-d5d9-4fe1-b9fc-66a15a162e41/export_locations -H "X-Openstack-Manila-API-Version: 2.9" -H "X-Auth-Token: $token" -H "Accept: application/json"`

URI

- GET /v2/{project_id}/shares/{share_id}/export_locations
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
export_locations	Array of strings	Specifies the export_location objects.

- Description of field **export_location**

Parameter	Type	Description
id	String	Specifies the ID of the mount location of the shared file system.
share_instance_id	String	Specifies the ID of the shared file system.
path	String	Specifies the path that will be used when the shared file system is mounted.
is_admin_only	Boolean	Specifies whether the shared file system is only visible to administrators and its owner. Possible values are true (only visible to administrators and its owner) and false (visible to all users).
preferred	Boolean	Identifies which mount locations are most efficient and are used preferentially when multiple mount locations exist.

- Example response

NFS file system:

```
{
  "export_locations": [
    {
      "path": "NFS:sfs-nas1.dong.com:/share-236b936a",
      "id": "b03d2aac-aeed-409a-af07-5d1b9024241c",
      "preferred": false
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.

8.2.6 Modifying a Shared File System

Function

This API is used to modify the name and description of a shared file system.

URI

- PUT /v2/{project_id}/shares/{share_id}
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
share	Yes	Object	Specifies the share objects.

- Description of the **share** field

Parameter	Mandatory	Type	Description
display_name	No	String	Specifies the new name of the shared file system. The value consists of 0 to 255 characters.
display_description	No	String	Describes the shared file system. The value contains 0 to 255 characters.
is_public	No	Boolean	(Supported by API v2.8 and later versions.) Specifies whether a file system can be publicly seen. If it is set to true , the file system can be seen publicly. If it is set to false , the file system can be seen privately. The default value is false .

- Example request

Modifying a shared file system (with the file system name changed to **testshare** and description to **test**):

```
{
  "share": {
    "display_name": "testshare",
    "display_description": "test"
  }
}
```

Response

- Parameter description

Parameter	Type	Description
share	Object	Specifies the share objects.

- Description of the **share** field

Parameter	Type	Description
links	Array	Specifies the links of shared file systems.
availability_zone	String	Specifies the availability zone.
share_server_id	String	Specifies the ID for managing share services.

Parameter	Type	Description
share_network_id	String	Specifies the ID of the share network. This parameter is reserved, because share network management is not supported currently.
snapshot_id	String	Specifies the ID of the source snapshot that is used to create the shared file system. This parameter is reserved, because snapshots are not supported currently.
snapshot_support	Boolean	Specifies whether snapshots are supported. This parameter is reserved, because snapshots are not supported currently.
id	String	Specifies the ID of the shared file system.
size	Integer	Specifies the size (GB) of the shared file system.
consistency_group_id	String	(Supported by API versions from v2.31 to v2.42.) Specifies the ID of a consistency group. This parameter is reserved, because consistency groups are not supported currently.
project_id	String	Specifies the ID of the project to which the shared file system belongs.
metadata	Object	Sets one or more metadata key and value pairs as a dictionary of strings. The value of the share_used key indicates the file system used capacity, in bytes. The value of the enterprise_project_id key indicates the ID of the enterprise project that the file system belongs to.
status	String	Specifies the status of the shared file system.
task_state	String	Specifies the data migration status. This parameter is reserved, because data migration is not supported currently.
has_replicas	Boolean	(Supported by API versions from v2.11 to v2.42.) Specifies whether any replication exists. This parameter is reserved, because replication is not supported currently.
replication_type	String	(Supported by API versions from v2.11 to v2.42.) Specifies the replication type. This parameter is reserved, because replication is not supported currently.

Parameter	Type	Description
description	String	Describes the shared file system.
host	String	Specifies the name of the host.
name	String	Specifies the name of the shared file system.
created_at	String	Specifies the date and time stamp when the shared file system was created.
access_rules_status	String	(Supported by API versions from v2.10 to v2.27.) Specifies the configuration status of the access rule. Possible values are active (effective), error (configuration failed), and syncing (configuration in progress).
share_proto	String	Specifies the protocol for sharing file systems.
volume_type	String	Specifies the volume type. The definition of this parameter is the same as that of share_type .
share_type_name	String	Specifies the storage service type assigned for the shared file system, such as high-performance storage (composed of SSDs) and large-capacity storage (composed of SATA disks).
share_type	String	Specifies the ID of the file system type.
export_locations	Array	Lists the mount locations. Currently, only a single mount location is supported. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.8 .
export_location	String	Specifies the mount location. This parameter exists only when X-Openstack-Manila-API-Version specified in the request header is smaller than 2.8 .
is_public	Boolean	(Supported by API versions from v2.8 to v2.42.) Specifies whether a file system can be publicly seen. If it is set to true , the file system can be seen publicly. If it is set to false , the file system can be seen privately. The default value is false .
source_share_group_snapshot_member_id	String	(Supported by API v2.31 and later versions.) Specifies the ID of a consistency snapshot source. Currently, the consistency group is not supported. This field is reserved.

Parameter	Type	Description
revert_to_snapshot_support	Boolean	(Supported by API v2.27 and later versions.) Specifies whether reversion to snapshot is supported. Currently, snapshot is not supported. This field is reserved.
create_share_from_snapshot_support	Boolean	(Supported by API v2.24 and later versions.) Specifies whether creating file systems from snapshot is supported. Currently, snapshot is not supported. This field is reserved.
mount_snapshot_support	Boolean	(Supported by API v2.32 and later versions.) Specifies whether snapshot mounting is supported. Currently, snapshot is not supported. This field is reserved.
user_id	String	(Supported by API v2.16 and later versions.) Specifies the user ID.

- Example response

```
{
  "share": {
    "status": "available",
    "share_type_name": "sla",
    "description": "test",
    "links": [
      {
        "href": "https://192.168.196.47:8796/v2/07412155bf474db9a2f697fd978593d7/shares/f26d867f-9876-433d-8db2-25d210f29309",
        "rel": "self"
      },
      {
        "href": "https://192.168.196.47:8796/07412155bf474db9a2f697fd978593d7/shares/f26d867f-9876-433d-8db2-25d210f29309",
        "rel": "bookmark"
      }
    ],
    "availability_zone": "az1.dc1",
    "share_network_id": null,
    "share_server_id": null,
    "share_group_id": null,
    "host": "DJ38@a4588256-3880-4136-b3c9-4c3aade8a84b#a4588256-3880-4136-b3c9-4c3aade8a84b",
    "revert_to_snapshot_support": null,
    "access_rules_status": "active",
    "snapshot_id": null,
    "create_share_from_snapshot_support": null,
    "is_public": true,
    "task_state": null,
    "snapshot_support": true,
    "id": "f26d867f-9876-433d-8db2-25d210f29309",
    "size": 1,
    "source_share_group_snapshot_member_id": null,
    "user_id": "daa3f8f8d7254465841da769298a76f6",
    "name": "manila share",
    "share_type": "8ae4e74e-83f4-4980-8ab8-e637f9294e0b",
    "has_replicas": false,
    "replication_type": null,
    "created_at": "2018-12-25T08:45:22.525899",
    "share_proto": "NFS",
    "volume_type": "sla",
  }
}
```

```

"mount_snapshot_support": null,
"project_id": "07412155bf474db9a2f697fd978593d7",
"metadata": {
  "share_key": "test",
  "share_used": "1",
}
}
}

```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.2.7 Deleting a Shared File System

Function

This API is used to delete a shared file system.

NOTE

This API is an asynchronous API. If the returned status code is **202**, the API request is successfully delivered and received. Later, you can query the shared file system by referring to [Querying Details About a Shared File System](#) to identify whether the deletion is complete and successful.

URI

- DELETE /v2/{project_id}/shares/{share_id}
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Example response
None

Status Codes

- Normal
202
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.3 File System Access Rules

8.3.1 Adding a File System Access Rule

Function

This API is used to add a file system access rule.

NOTE

- This API is an asynchronous API. If the returned status code is **200**, the API request is successfully delivered and received. Later, you can refer to [Querying File System Access Rules](#) to check whether the access rule is added successfully.

URI

- POST /v2/{project_id}/shares/{share_id}/action?
vpc_ip_base_acl={vpc_ip_base_acl}
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For how to obtain the project ID, see Obtaining a Project ID .
vpc_ip_base_acl	No	String	Specifies the identifier used with IP address-based authorization. Currently, only enable is available, which indicates that an access rule used with IP address-based authorization will be created. NOTICE To ensure compatibility, if this parameter is left blank or set to a value other than enable , you can still use this API to create an access rule used with IP address-based authorization. However, this method of creation has been discarded and will not be maintained in the future.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
os-allow_access	Yes	Object	Specifies the os-allow_access objects.

- Description of field **os-allow_access**

Parameter	Mandatory	Type	Description
access_level	No	String	Specifies the access level of the file system. Possible values are ro (read-only) and rw (read-write). The default value is rw (read-write).
access_type	Yes	String	Specifies the storage access method. <ul style="list-style-type: none"> If the NFS protocol is used, specify cert. If multiple protocols are used, specify cert. <p>Note</p> <ol style="list-style-type: none"> Value user indicates storage access using username. Value cert indicates storage access using VPC ID and IP address.

- Example request (IP address-based authorization)

POST /v2/{project_id}/shares/{share_id}/action?vpc_ip_base_acl=enable

Adding a file system access rule (value of the rule parameter

0560a527-0e77-40a6-

aa3b-110beecad368#127.0.0.1#1#all_squash,root_squash):

```
{
  "allow_access": {
    "access_to": "0560a527-0e77-40a6-aa3b-110beecad368#127.0.0.1#1#all_squash,root_squash",
    "access_type": "cert",
    "access_level": "rw"
  }
}
```

NOTICE

When creating the share access rule for an IP address-based authorization scenario.

- The **X-Openstack-Manila-API-Version** parameter must be specified for the request header, and the value of **X-Openstack-Manila-API-Version** must be from 2.28 to 2.42.

- The **vpc_ip_base_acl** parameter must be added in the request URL and the value of **vpc_ip_base_acl** must be set to **enable**. To ensure compatibility, if this parameter is left blank or set to a value other than **enable**, you can still use this API to create an access rule used with IP address-based authorization. However, this method of creation has been discarded and will not be maintained in the future.

Response

- Parameter description

Parameter	Type	Description
access	Object	Specifies the access objects. If the access rule is not updated, this value is null .

- Description of the **access** field

Parameter	Type	Description
share_id	String	Specifies the ID of the shared file system to which the access rule is added.
access_type	String	Specifies the type of the access rule.
access_to	String	Specifies the object that the backend grants or denies access.
access_level	String	Specifies the level of the access rule.
id	String	Specifies the ID of the access rule.
state	String	Specifies the status of the access rule. If the API version is earlier than 2.28, the status of the access rule is new , active , or error . In versions from 2.28 to 2.42, the status of the access rule is queued_to_apply , applying , active , error , queued_to_deny , or denying .

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.

Status Code	Description
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.3.2 Deleting a File System Access Rule

Function

This API is used to delete a file system access rule.

NOTE

This API is an asynchronous API. If the returned status code is **202**, the API request is successfully delivered and received. Later, you can refer to [Querying File System Access Rules](#) to identify whether the access rule is deleted successfully.

URI

- POST /v2/{project_id}/shares/{share_id}/action
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
os-deny_access	Yes	Object	Specifies the os-deny_access object.

- Description of field **os-deny_access**

Parameter	Mandatory	Type	Description
access_id	Yes	String	Specifies the ID of the access rule of the shared file system. The value contains 1 to 36 characters.

- Example request

Deleting a file system access rule (rule ID: **418e3cf4-08c3-4ed2-a29a-ceffa346b3b8**):

```
{
  "os-deny_access": {
    "access_id": "418e3cf4-08c3-4ed2-a29a-ceffa346b3b8"
  }
}
```

Response

- Parameter description

None

- Example response

None

Status Codes

- Normal
202
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.

Status Code	Description
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.3.3 Querying File System Access Rules

Function

This API is used to query the access rules of a shared file system.

URI

- POST /v2/{project_id}/shares/{share_id}/action
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
os-access_list	Yes	Object	Specifies the os-access_list object. To view access rules, set this value to null .

- Example request

```
{
  "os-access_list": null
}
```

Response

- Description

Parameter	Type	Description
access_list	Array of objects	Lists the access rules.

- Description of field **access_list**

Parameter	Type	Description
access_type	String	Specifies the type of the access rule.
access_to	String	Specifies the object that the backend grants or denies access.
access_level	String	Specifies the level of the access rule.

Parameter	Type	Description
state	String	Specifies the status of the access rule. If the API version is earlier than 2.28, the status of the access rule is new , active , or error . In versions from 2.28 to 2.42, the status of the access rule is queued_to_apply , applying , active , error , queued_to_deny , or denying .
id	String	Specifies the ID of the access rule.

- Example response

```
{
  "access_list": [
    {
      "access_level": "rw",
      "state": "active",
      "id": "85417bed-5e26-4c99-8c0c-92c95b5c640e",
      "access_type": "cert",
      "access_to": "a91556b7-c7c8-4273-915e-2729e04cdb01",
    },
    {
      "access_level": "rw",
      "state": "active",
      "id": "2ecbeb0b-b2ba-41f1-ba63-0666548925b9",
      "access_type": "cert",
      "access_to": "0560a527-0e77-40a6-aa3b-110beecad368#0.0.0.0/0#0#all_squash,root_squash",
      "created_at": "2017-07-07T03:15:06.858662",
      "updated_at": "2018-07-07T03:15:06.858662"
    },
    {
      "access_level": "rw",
      "state": "active",
      "id": "24615391-d58d-4a74-ac5a-520233c9c396",
      "access_type": "cert",
      "access_to": "0560a527-0e77-40a6-aa3b-110beecad368#192.168.196.47#1#all_squash,root_squash",
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.

Status Code	Description
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.4 Quota Management

Function

This API is used to query quota information.

URI

- GET /v2/{project_id}/os-quota-sets/{project_id}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the ID of the tenant whose quotas are to be queried, updated, or deleted. This ID differs from the first project ID (the administrative tenant ID) contained in the URI.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
quota_set	Object	Specifies the quota_set objects.

- Description of field **quota_set**

Parameter	Type	Description
gigabytes	Integer	Specifies the capacity in gigabytes allowed for each tenant.
snapshots	Integer	Specifies the number of snapshots allowed for each tenant.
shares	Integer	Specifies the number of shared file systems allowed for each tenant.
snapshot_gigabytes	Integer	Specifies the snapshot capacity in gigabytes allowed for each tenant.
id	String	Specifies the ID of the tenant for which you manage quotas.
share_networks	Integer	Specifies the number of share networks allowed for each tenant.

- Example response

```
{
  "quota_set": {
    "gigabytes": -1,
    "snapshots": -1,
    "snapshot_gigabytes": -1,
    "shares": -1,
    "id": "da0f615c35eb4d72812d1547a77b5394",
    "share_networks": 10
  }
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.5 Expansion and Shrinking

8.5.1 Expanding a Shared File System

Function

This API is used to expand the capacity of a shared file system.

NOTE

This API is an asynchronous API. If the returned status code is **202**, the API request is successfully delivered and received. Later, you can refer to [Querying Details About a Shared File System](#) to identify whether the shared file system is expanded successfully.

URI

- POST /v2/{project_id}/shares/{share_id}/action
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
os-extend	Yes	Object	Specifies the os-extend object.

- Description of field **os-extend**

Parameter	Mandatory	Type	Description
new_size	Yes	Integer	Specifies the new capacity (GB) of the shared file system.

- Example request

Expanding the capacity of a shared file system to 2 GB:

```
{
  "os-extend": {
    "new_size": 2
  }
}
```

Response

- Parameter description
None
- Example response
None

Status Codes

- Normal
202
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
400 Bad Request	Invalid input: The post-deduction capacity must be larger than 0 and smaller than the current capacity. (Current capacity: <i>XX</i> ; post-deduction capacity: <i>XX</i>)
400 Bad Request	Invalid input: The post-expansion capacity must be larger than the current capacity. (Current capacity: <i>XX</i> ; post-expansion capacity: <i>XX</i>)
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.

Status Code	Description
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.5.2 Shrinking a Shared File System

Function

This API is used to shrink the capacity of a shared file system.

NOTE

This API is an asynchronous API. If the returned status code is **202**, the API request is successfully delivered and received. Later, you can refer to [Querying Details About a Shared File System](#) to identify whether the shared file system is shrunk successfully.

URI

- POST /v2/{project_id}/shares/{share_id}/action
- Parameter description

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
os-shrink	Yes	Object	Specifies the os-shrink object.

- Description of field **os-shrink**

Parameter	Mandatory	Type	Description
new_size	Yes	Integer	Specifies the new capacity (GB) of the shared file system.

- Example request

Reducing the capacity of a shared file system to 1 GB:

```
{
  "os-shrink": {
    "new_size": 1
  }
}
```

Response

- Parameter description

None

- Example response

None

Status Codes

- Normal

202

- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.

Status Code	Description
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of an internal service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

8.6 Tag Management

8.6.1 Adding a Tag to a Shared File System

Function

This API is used to add a tag to a specified shared file system.

A shared file system can have a maximum of 10 tags.

The keys of multiple tags added to a shared file system must be unique.

This API is idempotent. If the key to be added has already been added to the shared file system, the tag is updated.

URI

- POST /v2/{project_id}/sfs/{share_id}/tags
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
share_id	Yes	String	Specifies the ID of the shared file system.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
tag	Yes	Resource_tag	Specifies the tag.

- Description of field **resource_tag**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. The value can contain a maximum of 36 characters. The key cannot be left blank and cannot contain non-printable ASCII characters (0-31) or the following characters: =* < > \, /
value	Yes	String	Specifies the tag value. The value contains a maximum of 43 characters and can be an empty string. It cannot contain non-printable ASCII characters (0-31) or the following characters: =* < > \, /

- Example request

Adding a tag (key1, value1) to a shared file system

```
{
  "tag": {
    "key": "key1",
    "value": "value1"
  }
}
```

Response

- Parameter description
None
- Example response
None

Status Codes

- Normal
204
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.2 Deleting a Tag from a Shared File System

Function

This API is used to delete a tag from a specified shared file system.

NOTE

If the key to be deleted does not exist in the shared file system, error 404 is returned after API calling.

URI

- DELETE /v2/{project_id}/sfs/{share_id}/tags/{key}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
share_id	Yes	String	Specifies the ID of the shared file system.

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Specifies the tag key. The value can contain a maximum of 36 characters. The key cannot be left blank and cannot contain non-printable ASCII characters (0-31) or the following characters: =*⟨⟩ \,/</p> <p>NOTICE When calling this API to delete a tag, if the tag key contains special characters that are not directly resolved by the URL, the key needs to be escaped.</p>

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description
None
- Example response
None

Status Codes

- Normal
204
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.

Status Code	Description
500 Internal Server Error	The request is not completed because of a service error.

8.6.3 Querying Tags of a Shared File System

Function

This API is used to query all tags of a specified shared file system.

URI

- GET /v2/{project_id}/sfs/{share_id}/tags
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
share_id	Yes	String	Specifies the ID of the shared file system.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
tags	Array of resource_tags	Specifies the list of tags.

Parameter	Type	Description
sys_tags	Array of resource_tags	Only the op_service permission can obtain this field. 1. Currently, only one resource_tag structure key is used, _sys_enterprise_project_id . 2. Currently, key contains only one value. 0 indicates the default enterprise project. This field cannot be returned in non-op_service scenarios.

- Description of field **resource_tag**

Parameter	Type	Description
key	String	Specifies the tag key.
value	String	Specifies the tag value.

- Example response

```
{
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": ""
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.4 Querying Tags of All File Systems of a Tenant

Function

This API is used to query the tags of all file systems of a tenant.

URI

- GET /v2/{project_id}/sfs/tags
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
tags	Array of tags	Specifies the list of tags.

- Description of the **tag** field

Parameter	Type	Description
key	String	Specifies the key of the tag.
values	Array of strings	Lists the values of the tag. The value is a list of tag values of all shared file systems of a tenant. Only one of the same tag values is displayed.

- Example response

```
{
  "tags": [ {
    "key": "key1",
    "values": [ "value1", "" ]
  }, {
    "key": "key2",
    "values": [ "value1", "value2" ]
  } ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.5 Batch Adding Tags to a Shared File System

Function

This API is used to batch add tags to a shared file system.

A shared file system can have a maximum of 10 tags.

The keys of multiple tags added to a shared file system must be unique.

This API is idempotent. If the key to be added has already been added to the shared file system, the tag is updated.

URI

- POST /v2/{project_id}/sfs/{share_id}/tags/action
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Parameter	Mandatory	Type	Description
share_id	Yes	String	Specifies the ID of the shared file system.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the operation identifier. Possible values are create and delete . Use create to batch add tags to a specified shared file system.
tags	No	Array of resource_tags	Specifies the tag list. This parameter is mandatory when the tenant permission is used. For the op_service permission, choose either this field or sys_tags .
sys_tags	No	Array of resource_tags	Specifies the system tag list. This field is available only to the op_service permission. Choose either this field or tags . Only one resource_tag structure key, _sys_enterprise_project_id , is used in TMS calls. The value is ID or 0 . Value 0 indicates the default enterprise project.

- Description of field **resource_tag**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. The value can contain a maximum of 36 characters. The key cannot be left blank and cannot contain non-printable ASCII characters (0-31) or the following characters: =*<>\, /

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the tag value. The value contains a maximum of 43 characters and can be an empty string. It cannot contain non-printable ASCII characters (0-31) or the following characters: =* < > \, /

- Example request

Batch adding tags (key1, value1 and key2, value2) to a shared file system

```
{
  "action": "create",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value2"
    }
  ]
}
```

Response

- Parameter description
None
- Example response
None

Status Codes

- Normal
204
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.6 Batch Deleting Tags from a Shared File System

Function

This API is used to batch delete tags from a specified shared file system.

This API is idempotent. If the tags to be deleted do not exist on the shared file system, the deletion is regarded as successful by default.

URI

- POST /v2/{project_id}/sfs/{share_id}/tags/action
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
share_id	Yes	String	Specifies the ID of the shared file system.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the operation identifier. Possible values are create and delete . Use delete to batch delete tags from a specified shared file system.
tags	Yes	Array of resource_tags	Specifies the tag list.

- Description of field **resource_tag**

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the tag key. The value can contain a maximum of 36 characters. This parameter cannot be left blank.
value	No	String	Specifies the tag value. The value contains a maximum of 43 characters and can be an empty string. If the value is not an empty string, the tag is located and deleted based on the key and value. If the value is an empty string, the tag is located and deleted based on the key.

- Example request

Batch deleting tags (key1, value1; key2; key3) from a shared file system

```
{
  "action": "delete",
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2"
    },
    {
      "key": "key3",
      "value": ""
    }
  ]
}
```

Response

- Parameter description
None
- Example response
None

Status Codes

- Normal
204
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.

Status Code	Description
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.7 Querying Shared File Systems by Tag

Function

This API is used to query shared file systems by tag.

URI

- POST /v2/{project_id}/sfs/resource_instances/action
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
offset	No	String	Specifies the index location. The value is a character string consisting of 0 and positive integers. The default value is 0. The first record in the query result is the offset+1 record that meets the query criteria.

Parameter	Mandatory	Type	Description
limit	No	String	Specifies the maximum number of query records. The value is a character string consisting of integers. The default value is 1000 . The value ranges from 1 to 1000 . The number of returned records cannot exceed the value of limit .
action	Yes	String	Specifies the operation identifier. Possible values are filter and count . Use filter to query details of a shared file system using tags.
matches	No	Array of matches	Specifies the file system query field. If this parameter is left null, all shared file systems of the tenant are searched by default.
tags	No	Array of tags	Specifies the tag search field, which is a list of tags. Only shared file systems containing all the listed tags can be returned. Tags in this search criteria are in the AND relationship. Specifically, a shared file system can be searched only when it meets all the tag search criteria. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of tags is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.

Parameter	Mandatory	Type	Description
tags_any	No	Array of tags	Specifies the tag search field, which is a list of tags. Shared file systems that contain any listed tag will be returned. Tags in this search criteria are in the OR relationship. Specifically, a shared file system can be searched as long as it meets one tag search condition. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of tags_any is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.
not_tags	No	Array of tags	Specifies the tag search field, which is a list of tags. Only shared file systems that contain none of the listed tags will be returned. Tags in this search criteria are in the NOR relationship. Specifically, a shared file system can be searched only when it does not meet any tag search criteria. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of not_tags is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.

Parameter	Mandatory	Type	Description
not_tags_any	No	Array of tags	Specifies the tag search field, which is a list of tags. Shared file systems that do not contain any of the listed tags will be returned. Tags in this search criteria are in the NAND relationship. Specifically, a shared file system can be searched as long as it does not meet one tag search condition. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of not_tags_any is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.
sys_tags	No	Array of tags	Only the op_service permission can use this field to filter resources. <ol style="list-style-type: none"> Currently, TMS can invoke only one tag structure key, _sys_enterprise_project_id. Currently, key contains only one value. 0 indicates the default enterprise project. sys_tags and tenant tag filtering conditions (tags, tags_any, not_tags, and not_tags_any) cannot be used at the same time.

NOTICE

In the request parameters, tag search fields **tags**, **tags_any**, **not_tags**, and **not_tags_any** are optional and can be combined with each other. Such tag search fields are in the AND relationship.

- Description of the **match** field

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the key. The value is fixed to resource_name .
value	Yes	String	Specifies the value. value indicates the name of a shared file system. An empty string specifies an exact match and only shared file systems whose names are empty can be queried. A non-empty string specifies a fuzzy query (case insensitive). The value can contain a maximum of 255 characters.

- Description of the **tag** field

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the key of the tag. A tag key can contain a maximum of 127 characters. This parameter cannot be left blank.
values	Yes	Array of strings	Lists the values. Each value can contain a maximum of 255 characters. If the value is left empty, any value is matched. The values are in the OR relationship.

- Example request

Querying shared file systems using tag (key1, value2)

```
{
  "offset": "0",
  "limit": "100",
  "action": "filter",
  "matches": [{
    "key": "resource_name",
    "value": "share_name"
  }],
  "tags": [{
    "key": "key1",
    "values": ["value2"]
  }, {
    "key": "key2",
    "values": []
  }],
  "tags_any": [{
    "key": "key3",
    "values": ["value3"]
  }, {
    "key": "key4",
    "values": []
  }],
  "not_tags": [{
    "key": "key5",
```

```

    "values": ["value5"]
  }, {
    "key": "key6",
    "values": []
  }
],
"not_tags_any": [{
  "key": "key7",
  "values": ["value7", "value8"]
}, {
  "key": "key9",
  "values": []
}]
}

```

- Example request (without passing **matches**)

```

{
  "offset": "0",
  "limit": "100",
  "action": "filter",
  "tags": [{
    "key": "key1",
    "values": ["value2"]
  }, {
    "key": "key2",
    "values": []
  }]
}

```

- Example request (without passing **limit** and **offset**)

```

{
  "action": "filter",
  "matches": [{
    "key": "resource_name",
    "value": "share_name"
  }],
  "tags": [{
    "key": "key1",
    "values": ["value2"]
  }, {
    "key": "key2",
    "values": []
  }]
}

```

- Example request (without passing **tags**, **not_tags**, **tags_any**, and **not_tags_any**)

```

{
  "offset": "0",
  "limit": "100",
  "action": "filter",
  "matches": [{
    "key": "resource_name",
    "value": "share_name"
  }]
}

```

- Example request (with the **action** field only)

```

{
  "action": "filter"
}

```

Response

- Parameter description

Parameter	Type	Description
resources	Array of resources	Specifies the list of shared file systems that meet the query criteria.
total_count	Integer	Specifies the total number of shared file systems that meet the query criteria. NOTE total_count specifies the total number of shared file systems that meet the query criteria, instead of the number returned after you set offset and limit .

- Data structure of the **resource** field

Parameter	Type	Description
resource_id	String	Specifies the ID of the shared file system.
resource_detail	Object	Specifies the resource details. The value is a resource object, used for extension. This value is left empty by default.
tags	Array of resource_tags	Specifies the list of tags. If no tags exist, the value is an empty array by default.
sys_tags	Array of tags	Only the op_service permission can obtain this field. 1. Currently, only one tag structure key is used, _sys_enterprise_project_id . 2. Currently, key contains only one value. 0 indicates the default enterprise project. This field cannot be returned in non-op_service scenarios.
resource_name	String	Specifies the resource name.

- Data structure of the **resource_tag** field

Parameter	Type	Description
key	String	Specifies the tag key. The value can contain a maximum of 36 characters. This parameter cannot be left blank. It cannot contain non-printable ASCII characters (0-31) or the following characters: =*<>\\, /

Parameter	Type	Description
value	String	Specifies the tag value. The value contains a maximum of 43 characters and can be an empty string. It cannot contain non-printable ASCII characters (0-31) or the following characters: =*<>\\,/

- Example response

```
{
  "resources":[
    {
      "resource_detail":{},
      "resource_id":"b1f3f06f-344d-446b-a4bf-647a225debae",
      "resource_name":"share_name",
      "tags":[
        {
          "key":"key1",
          "value": "value1"
        },
        {
          "key":"key2",
          "value": "value2"
        }
      ]
    }
  ],
  "total_count":1
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.6.8 Querying the Number of Shared File Systems by Tag

Function

This API is used to query the number of shared file systems by tag.

URI

- POST /v2/{project_id}/sfs/resource_instances/action
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description

Parameter	Mandatory	Type	Description
action	Yes	String	Specifies the operation identifier. Possible values are filter and count . Use count to query the number of share instances based on tags.
matches	No	Array of matches	Specifies the file system query field. If this parameter is left null, all shared file systems of the tenant are searched by default.

Parameter	Mandatory	Type	Description
tags	No	Array of tags	Specifies the tag search field, which is a list of tags. Only shared file systems containing all the listed tags can be returned. Tags in this search criteria are in the AND relationship. Specifically, a shared file system can be searched only when it meets all the tag search criteria. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of tags is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.
tags_any	No	Array of tags	Specifies the tag search field, which is a list of tags. Shared file systems that contain any listed tag will be returned. Tags in this search criteria are in the OR relationship. Specifically, a shared file system can be searched as long as it meets one tag search condition. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of tags_any is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.

Parameter	Mandatory	Type	Description
not_tags	No	Array of tags	Specifies the tag search field, which is a list of tags. Only shared file systems that contain none of the listed tags will be returned. Tags in this search criteria are in the NOR relationship. Specifically, a shared file system can be searched only when it does not meet any tag search criteria. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of not_tags is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.
not_tags_any	No	Array of tags	Specifies the tag search field, which is a list of tags. Shared file systems that do not contain any of the listed tags will be returned. Tags in this search criteria are in the NAND relationship. Specifically, a shared file system can be searched as long as it does not meet one tag search condition. In the key-values structure of each tag search condition, tag values are in the OR relationship. If the value of not_tags_any is not specified, all shared file systems meet the requirement of this tag search field. This field contains a maximum of 10 tag keys and each tag key has a maximum of 10 tag values. The tag value corresponding to each tag key can be an empty array but the structure cannot be missing. Tag keys must be unique. Tag values in a key-values structure must be unique.

Parameter	Mandatory	Type	Description
sys_tags	No	Array of tags	<p>Only the op_service permission can use this field to filter resources.</p> <ol style="list-style-type: none"> Currently, TMS can invoke only one tag structure key, _sys_enterprise_project_id. Currently, key contains only one value. 0 indicates the default enterprise project. sys_tags and tenant tag filtering conditions (tags, tags_any, not_tags, and not_tags_any) cannot be used at the same time.

NOTICE

In the request parameters, tag search fields **tags**, **tags_any**, **not_tags**, and **not_tags_any** are optional and can be combined with each other. Such tag search fields are in the AND relationship.

- Description of the **match** field

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the key. The value is fixed to resource_name .
value	Yes	String	Specifies the value. value indicates the name of a shared file system. An empty string specifies an exact match and only shared file systems whose names are empty can be queried. A non-empty string specifies a fuzzy query (case insensitive). The value can contain a maximum of 255 characters.

- Description of the **tag** field

Parameter	Mandatory	Type	Description
key	Yes	String	Specifies the key of the tag. A tag key can contain a maximum of 127 characters. This parameter cannot be left blank.

Parameter	Mandatory	Type	Description
values	Yes	Array of strings	Lists the values. Each value can contain a maximum of 255 characters. If the value is left empty, any value is matched. The values are in the OR relationship.

- Example request

Querying the number of shared file systems (file system name **share_name**) using tag (key1, value2)

```
{
  "action": "count",
  "matches": [{
    "key": "resource_name",
    "value": "share_name"
  }],
  "tags": [{
    "key": "key1",
    "values": ["value2"]
  }, {
    "key": "key2",
    "values": []
  }],
  "tags_any": [{
    "key": "key3",
    "values": ["value3"]
  }, {
    "key": "key4",
    "values": []
  }],
  "not_tags": [{
    "key": "key5",
    "values": ["value5"]
  }, {
    "key": "key6",
    "values": []
  }],
  "not_tags_any": [{
    "key": "key7",
    "values": ["value7", "value8"]
  }, {
    "key": "key9",
    "values": []
  }
]}
```

- Example request (without passing **matches**)

Querying the number of shared file systems using tag (key1, value2)

```
{
  "action": "count",
  "tags": [{
    "key": "key1",
    "values": ["value2"]
  }, {
    "key": "key2",
    "values": []
  }
]}
```

- Example request (without passing **tags**, **not_tags**, **tags_any**, and **not_tags_any**)

Querying the number of shared file systems (file system name **share_name**)

```
{
  "action": "count",
  "matches": [{
    "key": "resource_name",
    "value": "share_name"
  }]
}
```

- Example request (with the **action** field only)

Querying the total number of shared file systems of the tenant

```
{
  "action": "count"
}
```

Response

- Parameter description

Parameter	Type	Description
total_count	Integer	Specifies the total number of shared file systems that meet the query criteria.

- Example response

```
{
  "total_count": 1
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	Invalid value.
401 Unauthorized	Authentication failed.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested resource was not found.
500 Internal Server Error	The request is not completed because of a service error.

8.7 AZ

8.7.1 Querying Availability Zones

Function

This API is used to list availability zones (AZs).

NOTE

Query the API version. For details, see [Querying All API Versions](#). If the value of **version** is less than or equal to 2.6, add the request header **X-Openstack-Manila-API-Version: *Micro version number***, for example, **X-Openstack-Manila-API-Version: 2.4**.

URI

- GET /v2/{project_id}/availability-zones?share_az={share_az}
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	Specifies the project ID of the operator. For details about how to obtain the project ID, see Obtaining a Project ID .
share_az	No (query parameter)	Boolean	If this parameter is set to true , the current tenant can query AZs where shared file systems can be created. If this parameter is set to false , all AZs can be queried, regardless of whether shared file systems can be created there.

Request Header

The operation message header is the same as that of a common request. For details, see [Table 3-3](#).

Request

- Parameter description
None
- Example request
None

Response

- Parameter description

Parameter	Type	Description
availability_zones	Array of objects	List of availability zones

- Description of field **availability_zone**

Parameter	Type	Description
id	String	Specifies the ID of the AZ.
name	String	Specifies the AZ name.
created_at	String	Specifies the creation time of an AZ.
updated_at	String	Specifies the modification time of an AZ.

- Example response

```
{
  "availability_zones": [
    {
      "updated_at": null,
      "created_at": "2018-07-10T19:11:49.831107",
      "id": "dda1cb71-face-4526-a637-9768d19b20f3",
      "name": "az1.dc1"
    }
  ]
}
```

Status Codes

- Normal
200
- Abnormal

Status Code	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server could not be accepted by the client.

Status Code	Description
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	The request is not completed because of a service error.
501 Not Implemented	The request is not completed because the server does not support the requested function.
502 Bad Gateway	The request is not completed because the request is invalid.
503 Service Unavailable	The request is not completed because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

9 SFS Turbo APIs

9.1 Lifecycle Management

9.1.1 Creating a File System

Function

This API is used to create a file system.

URI

POST /v1/{project_id}/sfs-turbo/shares

Table 9-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 9-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-3 Request body parameters

Parameter	Mandatory	Type	Description
share	Yes	Share object	Request body for creating a file system
bss_param	No	BssInfo object	Yearly/Monthly billing mode. This parameter is mandatory.

Table 9-4 Share

Parameter	Mandatory	Type	Description
availability_zone	Yes	String	Code of the AZ where the file system resides
description	No	String	Description of the file system, which can contain 0 to 255 characters. This parameter is not supported by the current version.
enterprise_project_id	No	String	ID of the enterprise project to which the file system will be added
metadata	No	Metadata object	Metadata of the file system. The value consists of key and value pairs as a directory of strings.
name	Yes	String	Name of the SFS Turbo file system. The name contains 4 to 64 characters and must start with a letter. It can contain letters (case insensitive), digits, hyphens (-), and underscores (_), and cannot contain other special characters.
security_group_id	Yes	String	Security group ID of a tenant in a region
share_protocol	Yes	String	File sharing protocol. The valid value is NFS . Network File System (NFS) is a distributed file system protocol that allows different computers and operating systems to share data over a network.

Parameter	Mandatory	Type	Description
share_type	Yes	String	<p>File system type. Valid values are STANDARD and PERFORMANCE. This field is not returned when the file system is being created.</p> <ul style="list-style-type: none"> • For a previous-generation SFS Turbo file system, specify STANDARD for a Standard or Standard-Enhanced file system, and PERFORMANCE for a Performance or Performance-Enhanced file system. • For a 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, or 20 MB/s/TiB file system, this field is not verified. Specify either STANDARD or PERFORMANCE.

Parameter	Mandatory	Type	Description
size	Yes	Integer	<ul style="list-style-type: none"> For a previous-generation SFS Turbo file system, the capacity ranges from 500 to 32768, in GiB. For a previous-generation SFS Turbo file system with expand_type="bandwidth" configured under metadata, the capacity ranges from 10240 to 3276800, in GiB. For a 20 MB/s/TiB file system with expand_type="hpc" and hpc_bw="20M" configured under metadata, the capacity ranges from 3686 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system. For a 40 MB/s/TiB file system with expand_type="hpc" and hpc_bw="40M" configured under metadata, the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system. For a 125 MB/s/TiB file system with expand_type="hpc" and hpc_bw="125M"

Parameter	Mandatory	Type	Description
			<p>configured under metadata, the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system.</p> <ul style="list-style-type: none"> • For a 250 MB/s/TiB file system with expand_type="hpc" and hpc_bw="250M" configured under metadata, the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system. <p>- For a 500 MB/s/TiB file system with expand_type="hpc" and hpc_bw="500M" configured under metadata, the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system.</p> <p>- For a 1,000 MB/s/TiB file system with</p>

Parameter	Mandatory	Type	Description
			expand_type="hpc" and hpc_bw="1000M" configured under metadata , the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, specify 3686 GiB for a 3.6 TiB file system, 4915 GiB for a 4.8 TiB file system, and 8601 GiB for a 8.4 TiB file system.
subnet_id	Yes	String	Subnet ID of a tenant in a VPC
vpc_id	Yes	String	VPC ID of a tenant in a region
backup_id	No	String	Backup ID. This parameter is mandatory if you create a file system from a backup.
tags	No	Array of ResourceTag objects	Tag list

Table 9-5 Metadata

Parameter	Mandatory	Type	Description
crypt_key_id	No	String	ID of a KMS professional key. This parameter is used if you want to create an encrypted file system.
dedicated_flavor	No	String	VM flavor used for creating a dedicated file system
dedicated_storage_id	No	String	ID of the dedicated distributed storage used when creating a dedicated file system

Parameter	Mandatory	Type	Description
expand_type	No	String	<p>Extension type. This parameter is not returned when the file system is being created.</p> <p>This parameter is mandatory when you are creating an SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, 20 MB/s/TiB, or Enhanced file system.</p> <ul style="list-style-type: none"> Specify bandwidth when you are creating a Standard-Enhanced or Performance-Enhanced file system. Specify hpc when you are creating a 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, or 20 MB/s/TiB file system.
hpc_bw	No	String	<p>File system bandwidth.</p> <p>This parameter is mandatory when you are creating an SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, or 20 MB/s/TiB file system.</p> <p>Specify 20M for a 20 MB/s/TiB file system, 40M for a 40 MB/s/TiB file system, 125M for a 125 MB/s/TiB file system, and 250M for a 250 MB/s/TiB file system. Specify 500M for a 500 MB/s/TiB file system and 1000M for a 1,000 MB/s/TiB file system.</p>

Table 9-6 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key.</p> <p>It can contain a maximum of 128 characters.</p> <p>It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).</p>
value	Yes	String	<p>Tag value.</p> <p>Each tag value can contain a maximum of 255 characters and can be an empty string.</p> <p>It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).</p>

Table 9-7 BssInfo

Parameter	Mandatory	Type	Description
is_auto_renew	No	Long	Whether to enable automatic renewal
period_num	Yes	Long	Yearly/Monthly subscription terms
period_type	Yes	Long	Yearly/Monthly subscription type. The value can be 2 (monthly subscription) or 3 (yearly subscription).
is_auto_pay	No	Long	Whether to automatically pay for the order

Response Parameters

Status code: 202

Table 9-8 Response body parameters

Parameter	Type	Description
id	String	ID of the created SFS Turbo file system
name	String	Name of the created SFS Turbo file system
status	String	Status of the SFS Turbo file system

Example Requests

- Previous-generation SFS Turbo file system:

This example creates an SFS Turbo Standard file system in the AZ whose AZ code is **example**, with the file system name set to **sfs-turbo-test**, protocol type to NFS, capacity to 500 GB. The security group ID is **8c4ebbd0-6edf-4aae-8353-xxx**, the subnet ID is **b8884abe-f47b-4917-9f6c-xxx**, and the VPC ID is **d651ea2b-2b20-4c6d-8bbf-xxx**.

POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares

```
{
  "share": {
    "name": "sfs-turbo-test",
    "availability_zone": "example",
    "security_group_id": "8c4ebbd0-6edf-4aae-8353-xxx",
    "share_proto": "NFS",
    "share_type": "STANDARD",
    "size": 500,
    "subnet_id": "b8884abe-f47b-4917-9f6c-xxx",
    "vpc_id": "d651ea2b-2b20-4c6d-8bbf-xxx"
  }
}
```

- Previous-generation SFS Turbo file system in a dedicated scenario:

This example creates an SFS Turbo Standard file system in the AZ whose AZ code is **example**, with the file system name set to **sfs-turbo-dedicated-test**, protocol type to NFS, capacity to 500 GB. The dedicated storage pool ID is **198f0704-xxx-4d85-xxx-c25caa4d3264**, the dedicated ECS flavor is **c6.xlarge.2**, the security group ID is **8c4ebbd0-6edf-4aae-8353-xxx**, the subnet ID is **b8884abe-f47b-4917-9f6c-xxx**, and the VPC ID is **d651ea2b-2b20-4c6d-8bbf-xxx**.

POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares

```
{
  "share": {
    "name": "sfs-turbo-dedicated-test",
    "availability_zone": "example",
    "security_group_id": "8c4ebbd0-6edf-4aae-8353-xxx",
    "share_proto": "NFS",
    "share_type": "STANDARD",
    "size": 500,
    "subnet_id": "b8884abe-f47b-4917-9f6c-xxx",
    "vpc_id": "d651ea2b-2b20-4c6d-8bbf-xxx",
    "metadata": {
      "dedicated_flavor": "c6.xlarge.2",
    }
  }
}
```

```

        "dedicated_storage_id" : "198f0704-xxx-4d85-xxx-c25caa4d3264"
    }
}

```

- 125 MB/s/TiB:

This example creates an SFS Turbo 125 MB/s/TiB file system in the AZ whose AZ code is **example**, with the file system name set to **sfs-turbo-test**, protocol type to NFS, capacity to 3686 GB. The security group ID is **8c4ebbd0-6edf-4aae-8353-xxx**, the subnet ID is **b8884abe-f47b-4917-9f6c-xxx**, and the VPC ID is **d651ea2b-2b20-4c6d-8bbf-xxx**.

POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares

```

{
  "share" : {
    "name" : "sfs-turbo-test",
    "availability_zone" : "example",
    "security_group_id" : "8c4ebbd0-6edf-4aae-8353-xxx",
    "share_proto" : "NFS",
    "share_type" : "STANDARD",
    "size" : 3686,
    "subnet_id" : "b8884abe-f47b-4917-9f6c-xxx",
    "vpc_id" : "d651ea2b-2b20-4c6d-8bbf-xxx",
    "metadata" : {
      "expand_type" : "hpc",
      "hpc_bw" : "125M"
    }
  }
}

```

Example Responses

Status code: 202

Response body for creating a file system

```

{
  "id" : "708c017c-54b5-429a-a098-7692e23fa518",
  "name" : "sfs-turbo-test",
  "status" : "100"
}

```

Status Codes

Status Code	Description
202	Response body for creating a file system

Error Codes

See [Error Codes](#).

9.1.2 Querying Details About a File System

Function

This API is used to query details about an SFS Turbo file system.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}

Table 9-9 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-10 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-11 Response body parameters

Parameter	Type	Description
action_progress	ActionProgress object	Creation progress of the SFS Turbo file system. This field is only returned when the file system is being created.
version	String	Version of the SFS Turbo file system
avail_capacity	String	Available capacity of the SFS Turbo file system, in GB
availability_zone	String	Code of the AZ where the SFS Turbo file system resides
az_name	String	Name of the AZ where the SFS Turbo file system resides
created_at	String	Time when the file system was created. UTC time, for example: 2018-11-19T04:02:03
crypt_key_id	String	ID of the encryption key specified by the user. This parameter is not returned for non-encrypted file systems.

Parameter	Type	Description
expand_type	String	For an Enhanced file system, bandwidth is returned. For a 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, or 20 MB/s/TiB file system, hpc is returned. For other types of file systems, this field is not returned.
export_location	String	Location where the SFS Turbo file system is mounted. For example, 192.168.0.90:/. If the file system is being created, this parameter is not returned.
id	String	ID of the SFS Turbo file system
name	String	Name of the SFS Turbo file system specified during creation
pay_model	String	Billing mode of the SFS Turbo file system. Value 0 indicates pay-per-use. If the file system is being created, this field is not returned.
region	String	Region where the SFS Turbo file system resides
security_group_id	String	ID of the security group specified by the user
share_proto	String	Protocol used by the SFS Turbo file system. The valid value is NFS .
share_type	String	Storage class of the SFS Turbo file system. Valid values are STANDARD and PERFORMANCE .
size	String	Total capacity of the SFS Turbo file system, in GB
status	String	SFS Turbo file system status. The value can be as follows: 100 (creating), 200 (available), 303 (creation failed), and 800 (frozen)

Parameter	Type	Description
sub_status	String	<p>Sub-status of the SFS Turbo file system. The value can be as follows: This field is not returned if no modification is made to the file system.</p> <p>121 (expanding capacity), 132 (changing security group), 137 (adding authorized VPC), 138 (removing authorized VPC), 150* (adding storage backend), 151 (removing storage backend)</p> <p>221 (expansion succeeded), 232 (security group changed), 237 (authorized VPC added), 238 (authorized VPC removed), 250 (storage backend added), 257* (storage backend removed)</p> <p>321 (expansion failed), 332 (changing security group failed), 337 (adding authorized VPC failed), 338 (removing authorized VPC failed), 350 (adding storage backend failed), 351 (removing storage backend failed)</p>
subnet_id	String	ID of the subnet specified by the user
vpc_id	String	ID of the VPC specified by the user
enterprise_project_id	String	ID of the enterprise project to which the SFS Turbo file system is added
tags	Array of ResourceTag objects	Tag list
optional_endpoint	String	Alternative IP addresses that can be used for mounting. This field is not returned for previous-generation file systems.
hpc_bw	String	<p>File system bandwidth.</p> <ul style="list-style-type: none"> ● "20M": 20 MB/s/TiB ● "40M": 40 MB/s/TiB ● "125M": 125 MB/s/TiB ● "250M": 250 MB/s/TiB - "500M": 500 MB/s/TiB - "1000M": 1,000 MB/s/TiB
instanceId	String	Node ID of the file system type. This is a reserved field.
instanceType	String	Node type of the file system type. This is a reserved field.

Parameter	Type	Description
statusDetail	String	Request ID of the file system. This is a reserved field.
features	String	Whether backup is supported for SFS turbo file systems.

Table 9-12 ActionProgress

Parameter	Type	Description
CREATING	String	File system creation progress

Table 9-13 ResourceTag

Parameter	Type	Description
key	String	Tag key. It can contain a maximum of 128 characters. It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).
value	String	Tag value. Each tag value can contain a maximum of 255 characters and can be an empty string. It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).

Example Requests

Querying the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde
```

Example Responses

Status code: 200

Response body for querying a file system

```
{
  "id": "8fba8253-c914-439d-ae8b-d5c89d0bf5e8",
  "name": "sfs-turbo-8468",
  "status": "200",
  "version": "1.0.0",
  "region": "example",
  "availability_zone": "example",
  "az_name": "example",
  "created_at": "2018-11-19T04:02:03",
  "export_location": "192.168.xx.xx:/",
  "action_progress": { },
  "share_type": "STANDARD",
  "sub_status": "221",
  "vpc_id": "b24e39e1-bc0c-475b-ae0c-ae9cf240af3",
  "subnet_id": "86fc01ea-8ec8-409d-ba7a-e0ea16d4fd97",
  "security_group_id": "50586458-aec9-442c-bb13-e08ddc6f1b7a",
  "size": "600.00",
  "avail_capacity": "600.00",
  "pay_model": "0",
  "share_proto": "NFS"
}
```

Status Codes

Status Code	Description
200	Response body for querying a file system

Error Codes

See [Error Codes](#).

9.1.3 Deleting a File System

Function

This API is used to delete a file system.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}

Table 9-14 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-15 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

None

Example Requests

Deleting the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde
```

Example Responses

None

Status Codes

Status Code	Description
202	File system deletion request delivered.

Error Codes

See [Error Codes](#).

9.1.4 Obtaining the File System List

Function

This API is used to obtain the file system list.

URI

```
GET /v1/{project_id}/sfs-turbo/shares/detail
```


Table 9-16 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Table 9-17 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Long	Specifies the maximum number of returned file systems. If this parameter is not specified, the default value 1000 is used.
offset	No	Long	Sets the offset of the returned file system.

Request Parameters

Table 9-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-19 Response body parameters

Parameter	Type	Description
shares	Array of ShareInfo objects	List of SFS Turbo file systems
count	Integer	Number of SFS Turbo file systems

Table 9-20 ShareInfo

Parameter	Type	Description
action_progress	ActionProgress object	Creation progress of the SFS Turbo file system. This field is only returned when the file system is being created.
version	String	Version of the SFS Turbo file system
avail_capacity	String	Available capacity of the SFS Turbo file system, in GB
availability_zone	String	Code of the AZ where the SFS Turbo file system resides
az_name	String	Name of the AZ where the SFS Turbo file system resides
created_at	String	Time when the file system was created. UTC time, for example: 2018-11-19T04:02:03
crypt_key_id	String	ID of the encryption key specified by the user. This parameter is not returned for non-encrypted file systems.
expand_type	String	For an Enhanced file system, bandwidth is returned. For a 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, or 20 MB/s/TiB file system, hpc is returned. For other types of file systems, this field is not returned.
export_location	String	Location where the SFS Turbo file system is mounted. For example, 192.168.0.90:/. If the file system is being created, this parameter is not returned.
id	String	ID of the SFS Turbo file system
name	String	Name of the SFS Turbo file system specified during creation
pay_model	String	Billing mode of the SFS Turbo file system. Value 0 indicates pay-per-use. If the file system is being created, this field is not returned.
region	String	Region where the SFS Turbo file system resides
security_group_id	String	ID of the security group specified by the user
share_proto	String	Protocol used by the SFS Turbo file system. The valid value is NFS .
share_type	String	Storage class of the SFS Turbo file system. Valid values are STANDARD and PERFORMANCE .

Parameter	Type	Description
size	String	Total capacity of the SFS Turbo file system, in GB
status	String	SFS Turbo file system status. The value can be as follows: 100 (creating), 200 (available), 303 (creation failed), and 800 (frozen)
sub_status	String	Sub-status of the SFS Turbo file system. The value can be as follows: This field is not returned if no modification is made to the file system. 121 (expanding capacity), 132 (changing security group), 137 (adding authorized VPC), 138 (removing authorized VPC), 150* (adding storage backend), 151 (removing storage backend) 221 (expansion succeeded), 232 (security group changed), 237 (authorized VPC added), 238 (authorized VPC removed), 250 (storage backend added), 257* (storage backend removed) 321 (expansion failed), 332 (changing security group failed), 337 (adding authorized VPC failed), 338 (removing authorized VPC failed), 350 (adding storage backend failed), 351 (removing storage backend failed)
subnet_id	String	ID of the subnet specified by the user
vpc_id	String	ID of the VPC specified by the user
enterprise_project_id	String	ID of the enterprise project to which the SFS Turbo file system is added
tags	Array of ResourceTag objects	Tag list
optional_endpoint	String	Alternative IP addresses that can be used for mounting. This field is not returned for previous-generation file systems.
hpc_bw	String	File system bandwidth. <ul style="list-style-type: none"> • "20M": 20 MB/s/TiB • "40M": 40 MB/s/TiB • "125M": 125 MB/s/TiB • "250M": 250 MB/s/TiB - "500M": 500 MB/s/TiB - "1000M": 1,000 MB/s/TiB

Parameter	Type	Description
instanceId	String	Node ID of the file system type. This is a reserved field.
instanceType	String	Node type of the file system type. This is a reserved field.
statusDetail	String	Request ID of the file system. This is a reserved field.
features	String	Whether backup is supported for SFS turbo file systems.

Table 9-21 ActionProgress

Parameter	Type	Description
CREATING	String	File system creation progress

Table 9-22 ResourceTag

Parameter	Type	Description
key	String	Tag key. It can contain a maximum of 128 characters. It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).
value	String	Tag value. Each tag value can contain a maximum of 255 characters and can be an empty string. It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).

Example Requests

Querying file systems in the project whose ID
e1e45b08f3ea4480ab4655ef9c7160ba

GET [HTTPS://\[endpoint\]/v1/e1e45b08f3ea4480ab4655ef9c7160ba/sfs-turbo/shares/detail](https://[endpoint]/v1/e1e45b08f3ea4480ab4655ef9c7160ba/sfs-turbo/shares/detail)

Example Responses

Status code: 200

Response body for querying the file system list

```
{
  "shares" : [ {
    "id" : "8fba8253-c914-439d-ae8b-d5c89d0bf5e8",
    "name" : "sfs-turbo-8468",
    "status" : "100",
    "version" : "1.0.0",
    "region" : "example",
    "created_at" : "2018-11-19T04:02:03",
    "action_progress" : {
      "CREATING" : "22%"
    }
  },
  "share_type" : "STANDARD",
  "sub_status" : "",
  "availability_zone" : "az1.dc1",
  "az_name" : "az1",
  "vpc_id" : "b24e39e1-bc0c-475b-ae0c-ae9cf240af3",
  "subnet_id" : "86fc01ea-8ec8-409d-ba7a-e0ea16d4fd97",
  "security_group_id" : "50586458-aec9-442c-bb13-e08ddc6f1b7a",
  "size" : "500.00",
  "pay_model" : "0",
  "avail_capacity" : "500.00",
  "share_proto" : "NFS"
}, {
  "id" : "65f2d30b-7b4e-4786-9608-4324faef6646",
  "name" : "sfs-turbo-df12",
  "status" : "200",
  "version" : "1.0.0",
  "region" : "example",
  "created_at" : "2018-11-15T02:32:10",
  "export_location" : "192.168.xx.xx:/",
  "optional_endpoint" : "192.168.xx.xx 192.168.xx.xx",
  "share_type" : "HPC_PERFORMANCE_250M",
  "expand_type" : "hpc",
  "sub_status" : "",
  "availability_zone" : "az1.dc1",
  "az_name" : "az1",
  "vpc_id" : "b24e39e1-bc0c-475b-ae0c-ae9cf240af3",
  "subnet_id" : "86fc01ea-8ec8-409d-ba7a-e0ea16d4fd97",
  "security_group_id" : "50586458-aec9-442c-bb13-e08ddc6f1b7a",
  "size" : "3686.00",
  "pay_model" : "0",
  "avail_capacity" : "3686.00",
  "share_proto" : "NFS"
} ]
}
```

Status Codes

Status Code	Description
200	Response body for querying the file system list

Error Codes

See [Error Codes](#).

9.1.5 Expanding the Capacity of a File System

Function

This API is used to expand the capacity of a file system.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/action

Table 9-23 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-24 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-25 Request body parameters

Parameter	Mandatory	Type	Description
extend	Yes	Extend object	Object of extend

Table 9-26 Extend

Parameter	Mandatory	Type	Description
new_size	Yes	Integer	<p>New capacity of the file system, in GiB</p> <p>For a previous-generation Standard or Performance file system, the capacity ranges from 500 to 32768 (in GiB), and the expansion increment is 100 GiB.</p> <p>For a previous-generation Standard-Enhanced or Performance-Enhanced file system, the capacity ranges from 10240 to 327680 (in GiB), and the expansion increment is 100 GiB.</p> <p>For a 20 MB/s/TiB file system, the capacity ranges from 3686 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, use 4915 GiB for a 4.8 TiB file system and 8601 GiB for a 8.4 TiB file system. The expansion increment is 1.2 TiB.</p> <p>For a 40 MB/s/TiB file system, the capacity ranges from 1228 to 1048576 (in GiB) and must be a multiple of 1.2 TiB. The desired capacity must be converted to GiB and rounded down to the nearest integer. For example, use 4915 GiB for a 4.8 TiB file system and 8601 GiB for a 8.4 TiB file system. The expansion increment is 1.2 TiB.</p> <p>The capacity range and expansion increment of 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB and 125 MB/s/TiB file systems are the same as those of 40 MB/s/TiB file systems.</p>

Parameter	Mandatory	Type	Description
new_bandwidth	No	Long	New bandwidth, in GB. Only HPC Cache file systems support bandwidth change. The following bandwidths are supported: 2G, 4G, 8G, 16G, 24G, 32G and 48G .
bss_param	No	BssInfoExtended object	Expansion billing details of yearly/monthly resources

Table 9-27 BssInfoExtend

Parameter	Mandatory	Type	Description
is_auto_pay	No	Long	Whether to enable automatic payment

Response Parameters

Status code: 202

Table 9-28 Response body parameters

Parameter	Type	Description
id	String	ID of the SFS Turbo file system
name	String	Name of the SFS Turbo file system

Example Requests

Expanding the capacity of a file system to 1,000 GB

```
{
  "extend" : {
    "new_size" : 1000
  }
}
```

Example Responses

Status code: 202

Response body for expanding the capacity of a file system

```
{
  "id" : "67d4bd5e-7b2f-4c24-9a0b-c0038940c6f8",
  "name" : "sfs-turbo-test"
}
```


Status Codes

Status Code	Description
202	Response body for expanding the capacity of a file system

Error Codes

See [Error Codes](#).

9.2 Connection Management

9.2.1 Changing the Security Group Associated with a File System

Function

This API is used to change the security group associated with an SFS Turbo file system. Security group change is an asynchronous task. You can check whether the security group is changed based on the value of **sub_status** returned after calling the API to query details of a file system. If value **232** is returned, the security group has been changed.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/action

Table 9-29 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-30 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-31 Request body parameters

Parameter	Mandatory	Type	Description
change_security_group	Yes	ChangeSecurityGroup object	Object of change_security_group

Table 9-32 ChangeSecurityGroup

Parameter	Mandatory	Type	Description
security_group_id	Yes	String	ID of the security group to be changed

Response Parameters

Status code: 202

Table 9-33 Response body parameters

Parameter	Type	Description
id	String	ID of the SFS Turbo file system

Example Requests

Changing the security group of a file system (target security group ID **26f6b565-240e-43c3-8867-03f0bd975433**)

```
{
  "change_security_group": {
    "security_group_id": "26f6b565-240e-43c3-8867-03f0bd975433"
  }
}
```

Example Responses

Status code: 202

ID of the SFS Turbo file system

```
{
  "id": "67d4bd5e-7b2f-4c24-9a0b-c0038940c6f8"
}
```

Status Codes

Status Code	Description
202	ID of the SFS Turbo file system

Error Codes

See [Error Codes](#).

9.3 Tag Management

9.3.1 Adding a Tag for a File System

Function

This API is used to add a tag to a specified file system.

A maximum of 20 tags can be added to a file system.

Tag keys added to the same file system must be unique.

This API is idempotent. If the file system already has the key you want to add, the tag will be updated.

URI

POST /v1/{project_id}/sfs-turbo/{share_id}/tags

Table 9-34 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-35 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-36 Request body parameters

Parameter	Mandatory	Type	Description
tag	Yes	ResourceTag object	Description of the <code>resource_tag</code> field

Table 9-37 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. It can contain a maximum of 128 characters. It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).
value	Yes	String	Tag value. Each tag value can contain a maximum of 255 characters and can be an empty string. It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).

Response Parameters

None

Example Requests

Creating a file system tag, with tag value set to **key1** and tag key **value1**

```
{
  "tag": {
    "key": "key1",
    "value": "value1"
  }
}
```

```
}  
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Tag adding request delivered.

Error Codes

See [Error Codes](#).

9.3.2 Querying Tags of a File System

Function

This API is used to query all tags of a specified file system.

URI

GET /v1/{project_id}/sfs-turbo/{share_id}/tags

Table 9-38 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-39 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-40 Response body parameters

Parameter	Type	Description
tags	Array of ResourceTag objects	Tag list
sys_tags	Array of ResourceTag objects	<p>Only users with the op_service permission can obtain this field.</p> <ol style="list-style-type: none"> 1. This field currently contains only one resource_tag structure key, _sys_enterprise_project_id. 2. The key contains only value 0 currently, which indicates the default enterprise project. <p>This field is not returned for users without the op_service permission.</p>

Table 9-41 ResourceTag

Parameter	Type	Description
key	String	<p>Tag key.</p> <p>It can contain a maximum of 128 characters. It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).</p>
value	String	<p>Tag value.</p> <p>Each tag value can contain a maximum of 255 characters and can be an empty string. It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).</p>

Example Requests

Querying tags of the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/v1/{project_id}/sfs-turbo/77ba6f4b-6365-4895-8dda-bc7142af4dde/tags
```

Example Responses

Status code: 200

Response body for query all tags of a specified file system

```
{
  "tags": [ {
    "key": "key1",
    "value": "value1"
  }, {
    "key": "key2",
    "value": "value1"
  } ]
}
```

Status Codes

Status Code	Description
200	Response body for query all tags of a specified file system

Error Codes

See [Error Codes](#).

9.3.3 Deleting a Tag of a File System

Function

This API is used to delete a tag of a specified file system. If the key to be deleted does not exist, error 404 will be returned.

URI

DELETE /v1/{project_id}/sfs-turbo/{share_id}/tags/{key}

Table 9-42 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Parameter	Mandatory	Type	Description
key	Yes	String	<p>Tag key, which can contain a maximum of 128 characters. It cannot be left blank and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).</p> <p>When this API is called to delete a tag, if the tag key contains special characters that cannot be directly resolved by the URL, the key needs to be escaped.</p>

Request Parameters

Table 9-43 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

None

Example Requests

Deleting tags whose key is **test** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/77ba6f4b-6365-4895-8dda-bc7142af4dde/tags/test
```

Example Responses

None

Status Codes

Status Code	Description
204	File system tag deleted.

Error Codes

See [Error Codes](#).

9.3.4 Batch Adding Tags to a File System

Function

This API is used to batch add tags for a specified file system.

A maximum of 20 tags can be added to a file system.

Tag keys added to the same file system must be unique.

This API is idempotent. If the file system already has the key you want to add, the tag will be updated.

URI

POST /v1/{project_id}/sfs-turbo/{share_id}/tags/action

Table 9-44 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-45 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-46 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Operation identifier. The value is create . Use create if you want to batch add tags to a file system.
tags	No	Array of ResourceTag objects	Tag list. This field is mandatory for users. For users with the op_service permission, choose either this field or sys_tags .
sys_tags	No	Array of ResourceTag objects	System tag list. This field is available only to users with the op_service permission. Choose either this field or tags . Only one resource_tag structure key, _sys_enterprise_project_id , is used in TMS calls.

Table 9-47 ResourceTag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. It can contain a maximum of 128 characters. It cannot be left empty and cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).

Parameter	Mandatory	Type	Description
value	Yes	String	Tag value. Each tag value can contain a maximum of 255 characters and can be an empty string. It cannot contain the following characters: ASCII (0-31), equal signs (=), asterisks (*), left angle brackets (<), right angle brackets (>), backslashes (\), commas (,), vertical bars (), and slashes (/). It can contain only letters, digits, hyphens (-), and underscores (_).

Response Parameters

None

Example Requests

Batch adding tags for a file system, with tag key of the first tag set to **key1**, tag value of the first tag **value1**, tag key of the second tag **key2**, and tag value of the second tag **value1**

```
{
  "action": "create",
  "tags": [ {
    "key": "key1",
    "value": "value1"
  }, {
    "key": "key2",
    "value": "value1"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
204	File system tags added.

Error Codes

See [Error Codes](#).

9.3.5 Querying Tags of All File Systems of a Tenant

Function

This API is used to query the tags of all file systems of a tenant.

URI

GET /v1/{project_id}/sfs-turbo/tags

Table 9-48 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Table 9-49 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of returned tags
offset	No	Integer	Tag query offset

Request Parameters

Table 9-50 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-51 Response body parameters

Parameter	Type	Description
tags	Array of Tag objects	Tag list

Table 9-52 Tag

Parameter	Type	Description
key	String	Tag key. A key can contain a maximum of 128 characters and cannot be left blank.
values	Array of strings	Tag values. Each value can contain a maximum of 255 characters. An empty list of values can match with any value. All values of a tag key are in the OR relationship.

Example Requests

Query tags of all file systems in the project whose ID is **e1e45b08f3ea4480ab4655ef9c7160ba**

```
GET HTTPS://{endpoint}/v1/e1e45b08f3ea4480ab4655ef9c7160ba/sfs-turbo/tags
```

Example Responses

Status code: 200

Response body for querying a file system

```
{
  "tags": [ {
    "key": "key1",
    "values": [ "value1", "" ]
  }, {
    "key": "key2",
    "values": [ "value1", "value2" ]
  } ]
}
```

Status Codes

Status Code	Description
200	Response body for querying a file system

Error Codes

See [Error Codes](#).

9.4 Name Management

9.4.1 Changing the Name of a File System

Function

This API is used to change the name of an SFS Turbo file system.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/action

Table 9-53 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-54 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-55 Request body parameters

Parameter	Mandatory	Type	Description
change_name	Yes	ShareName object	SFS Turbo file system to be modified

Table 9-56 ShareName

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the SFS Turbo file system to be modified

Response Parameters

None

Example Requests

Changing the name of an SFS Turbo file system to **sfs-turbo-test1**

```
{
  "change_name": {
    "name": "sfs-turbo-test1"
  }
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Request successful
400	Invalid parameter
409	The file system name already exists.
500	Internal error

Error Codes

See [Error Codes](#).

9.5 File System Management

9.5.1 Creating an Asynchronous Task for a File System

Function

This API is used to create an asynchronous task for a file system. Only tasks for querying directory usage can be created. Such tasks are also referred to as DU tasks. The value of **feature** in the API request path is **dir-usage**.

Constraints

This API is only supported for file systems created after August 1, 2023. If there are 10 tasks being executed, no more task can be created. You are advised not to submit five or more requests at a time, or the file system performance may be affected. It takes a long time to query an oversized directory. Do not submit the request repeatedly. The value of **feature** in the API request path can only be the following:

- dir-usage

This API is only supported for the following types of file systems:

- 20 MB/s/TiB
- 40 MB/s/TiB
- 125 MB/s/TiB
- 250 MB/s/TiB
- 500 MB/s/TiB
- 1,000 MB/s/TiB

For SFS Turbo Standard, Standard-Enhanced, Performance, Performance-Enhanced file systems, use the API for querying the file system directory usage.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/{feature}/tasks

Table 9-57 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
feature	Yes	String	Task type. Only dir-usage is supported currently.

Request Parameters

Table 9-58 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type, which can be application or json

Table 9-59 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of a directory in the file system. The length of a single-level directory cannot exceed 255 characters, and that of a full path cannot exceed 4,096 characters.

Response Parameters

Status code: 202

Table 9-60 Response body parameters

Parameter	Type	Description
task_id	String	Task ID

Status code: 400

Table 9-61 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-62 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

DU task:

Creating a DU task for querying the usage of a directory. **path** is used to specify the full path of a valid directory. The length of a single-level directory cannot exceed 255 characters, and that of a full path cannot exceed 4,096 characters.

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-usage/tasks
{
  "path" : "/path"
}
```

Example Responses

Status code: 202

Accepted

```
{
  "task_id" : "d651ea2b-2b20-4c6d-8bbf-2adcec18dac9"
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0123",
  "errMsg" : "feature invalid"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "internal server error"
}
```

Status Codes

Status Code	Description
202	Accepted
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.5.2 Obtaining Asynchronous Tasks of a File System

Function

This API is used to obtain the asynchronous tasks of a file system. Only tasks for querying directory usage can be obtained. Such tasks are also referred to as DU tasks. The value of **feature** in the API request path is **dir-usage**.

Constraints

This API is only supported for file systems created after August 1, 2023. The obtained data may not be the latest as there is a 5-minute delay between the frontend and backend. The value of **feature** in the API request path can only be the following:

- dir-usage

This API is only supported for the following types of file systems:

- 20 MB/s/TiB

- 40 MB/s/TiB
- 125 MB/s/TiB
- 250 MB/s/TiB
- 500 MB/s/TiB
- 1,000 MB/s/TiB

For SFS Turbo Standard, Standard-Enhanced, Performance, Performance-Enhanced file systems, use the API for querying the file system directory usage.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/{feature}/tasks

Table 9-63 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
feature	Yes	String	Task type. For example, the value for DU tasks is dir-usage .

Table 9-64 Query Parameters

Parameter	Mandatory	Type	Description
marker	No	String	Marker. The value is the task ID.
limit	No	Integer	Limit. The value must be a positive integer. The default value is 20 and the maximum value is 100 .

Request Parameters

Table 9-65 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type, which can be application or json

Response Parameters

Status code: 200

Table 9-66 Response body parameters

Parameter	Type	Description
tasks	Array of OneFsTaskResp objects	Task list

Table 9-67 OneFsTaskResp

Parameter	Type	Description
task_id	String	Task ID
status	String	Task status, which can be SUCCESS , DOING , or FAIL .
dir_usage	FsDuInfo object	Resource usages of a directory (subdirectories included)
begin_time	String	Task start time in UTC format, for example, 2006-01-02 15:04:05
end_time	String	Task end time in UTC format, for example, 2006-01-02 15:04:06

Table 9-68 FsDuInfo

Parameter	Type	Description
path	String	Valid full path of a directory in the file system
used_capacity	Long	Used capacity, in byte
file_count	FsFileCount object	Total number of files in this directory
message	String	Error message

Table 9-69 FsFileCount

Parameter	Type	Description
dir	Long	Number of directories
regular	Long	Number of common files

Parameter	Type	Description
pipe	Long	Number of pipe files
char	Long	Number of character devices
block	Long	Number of block devices
socket	Long	Number of sockets
symlink	Long	Number of symbolic links

Status code: 400

Table 9-70 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 404

Table 9-71 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-72 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Obtaining 50 tasks starting from the task whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/dir-usage/tasks?marker=11abef677ac40f46644d1d5cfc2424a4&limit=50
```

Example Responses

Status code: 200

Successful

```
{
  "tasks": [ {
    "task_id": "2b31ed520xxxxxebedb6e57xxxxxxx",
    "status": "SUCCESS",
    "dir_usage": {
      "path": "/path",
      "used_capacity": 0,
      "file_count": {
        "dir": 0,
        "regular": 0,
        "pipe": 0,
        "char": 0,
        "block": 0,
        "socket": 0,
        "symlink": 0
      },
      "message": ""
    },
    "begin_time": "2023-03-01 11:46:01",
    "end_time": "2023-03-01 11:46:01"
  } ]
}
```

Status code: 400

Error response

```
{
  "errCode": "SFS.TURBO.0123",
  "errMsg": "feature invalid"
}
```

Status code: 404

Error response

```
{
  "errCode": "SFS.TURBO.0124",
  "errMsg": "task_id not found"
}
```

Status code: 500

Error response

```
{
  "errCode": "SFS.TURBO.0005",
  "errMsg": "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful

Status Code	Description
400	Error response
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.5.3 Obtaining Details About a File System Asynchronous Task

Function

This API is used to obtain details about a file system asynchronous task. Only tasks for querying directory usage can be obtained. Such tasks are also referred to as DU tasks. The value of **feature** in the API request path is **dir-usage**.

Constraints

This API is only supported for file systems created after August 1, 2023. The obtained data may not be the latest as there is a 5-minute delay between the frontend and backend. The value of **feature** in the API request path can only be the following:

- dir-usage

This API is only supported for the following types of file systems:

- 20 MB/s/TiB
- 40 MB/s/TiB
- 125 MB/s/TiB
- 250 MB/s/TiB
- 500 MB/s/TiB
- 1,000 MB/s/TiB

For SFS Turbo Standard, Standard-Enhanced, Performance, Performance-Enhanced file systems, use the API for querying the file system directory usage.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/{feature}/tasks/{task_id}

Table 9-73 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
feature	Yes	String	Task type. For example, the value for DU tasks is dir-usage .
task_id	Yes	String	Task ID

Request Parameters

Table 9-74 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type, which can be application or json

Response Parameters

Status code: 200

Table 9-75 Response body parameters

Parameter	Type	Description
task_id	String	Task ID
status	String	Task status, which can be SUCCESS , DOING , or FAIL .
dir_usage	FsDuInfo object	Resource usages of a directory (subdirectories included)
begin_time	String	Task start time in UTC format, for example, 2006-01-02 15:04:05
end_time	String	Task end time in UTC format, for example, 2006-01-02 15:04:06

Table 9-76 FsDulInfo

Parameter	Type	Description
path	String	Valid full path of a directory in the file system
used_capacity	Long	Used capacity, in byte
file_count	FsFileCount object	Total number of files in this directory
message	String	Error message

Table 9-77 FsFileCount

Parameter	Type	Description
dir	Long	Number of directories
regular	Long	Number of common files
pipe	Long	Number of pipe files
char	Long	Number of character devices
block	Long	Number of block devices
socket	Long	Number of sockets
symlink	Long	Number of symbolic links

Status code: 400

Table 9-78 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 404

Table 9-79 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-80 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Obtaining details of the task whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/fs/dir-usage/tasks/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 200

Successful

```
{
  "task_id": "2b31ed520xxxxxxebdb6e57xxxxxxx",
  "status": "SUCCESS",
  "dir_usage": {
    "path": "/path",
    "used_capacity": 0,
    "file_count": {
      "dir": 0,
      "regular": 0,
      "pipe": 0,
      "char": 0,
      "block": 0,
      "socket": 0,
      "symlink": 0
    },
    "message": ""
  },
  "begin_time": "2023-03-01 11:46:01",
  "end_time": "2023-03-01 11:46:01"
}
```

Status code: 400

Error response

```
{
  "errCode": "SFS.TURBO.0123",
  "errMsg": "feature invalid"
}
```

Status code: 404

Error response

```
{
  "errCode": "SFS.TURBO.0124",
  "errMsg": "task_id not found"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful
400	Error response
404	Error response
500	Error response

Error CodesSee [Error Codes](#).

9.5.4 Canceling or Deleting an Asynchronous Task of a File System

Function

If the asynchronous task is being executed, this API is used to cancel and then delete the task. Otherwise, this API is used to delete the task. Only tasks for querying directory usage can be deleted. Such tasks are also referred to as DU tasks. The value of **feature** in the API request path is **dir-usage**.

Constraints

This API is only supported for file systems created after August 1, 2023. The value of **feature** in the API request path can only be the following:

- dir-usage

This API is only supported for the following types of file systems:

- 20 MB/s/TiB
- 40 MB/s/TiB
- 125 MB/s/TiB
- 250 MB/s/TiB
- 500 MB/s/TiB
- 1,000 MB/s/TiB

For SFS Turbo Standard, Standard-Enhanced, Performance, Performance-Enhanced file systems, use the API for querying the file system directory usage.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/{feature}/tasks/{task_id}

Table 9-81 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
feature	Yes	String	Task type. For example, the value for DU tasks is dir-usage .
task_id	Yes	String	Task ID

Request Parameters

Table 9-82 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type, which can be application or json

Response Parameters

Status code: 400

Table 9-83 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 404

Table 9-84 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-85 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting the task whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/
dir-usage/tasks/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0123",
  "errMsg" : "feature invalid"
}
```

Status code: 404

Error response

```
{
  "errCode" : "SFS.TURBO.0124",
  "errMsg" : "task_id not found"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
202	Accepted
400	Error response
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6 Storage Interworking Management

9.6.1 Adding a Backend Target

Function

This API is used to add a storage backend for an SFS Turbo file system.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems. Request body parameter **file_system_path** must be the name of a directory that cannot be found in the root directory of the file system.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/targets

Table 9-86 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-87 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-88 Request body parameters

Parameter	Mandatory	Type	Description
file_system_path	Yes	String	Name of the linkage directory. SFS Turbo creates a subdirectory with this name in the root directory of the file system. The directory is used to bind the backend storage. The value must be a directory name that does not exist in the root directory of the file system. The value contains a maximum of 63 characters. The subdirectory name cannot be a period (.) or two consecutive periods (..). Multi-level directories are not supported, and slashes (/) are not allowed.
obs	Yes	ObsDataRepository object	OBS target

Table 9-89 ObsDataRepository

Parameter	Mandatory	Type	Description
bucket	Yes	String	OBS bucket name
endpoint	Yes	String	Name of the region where the bucket belongs
policy	No	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend

Parameter	Mandatory	Type	Description
attributes	No	ObsTargetAttributes object	Properties of the storage backend. This parameter is not supported for file systems that are created on or before June 30, 2024 and have not been upgraded. Submit a service ticket if you need it.

Table 9-90 ObsDataRepositoryPolicy

Parameter	Mandatory	Type	Description
auto_export_policy	No	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-91 AutoExportPolicy

Parameter	Mandatory	Type	Description
events	No	Array of strings	<p>Type of data automatically exported to the OBS bucket.</p> <ul style="list-style-type: none"> • NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Table 9-92 ObsTargetAttributes

Parameter	Mandatory	Type	Description
file_mode	No	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Mandatory	Type	Description
dir_mode	No	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	No	Integer	<p>ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).</p>

Parameter	Mandatory	Type	Description
gid	No	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Response Parameters

Status code: 202

Table 9-93 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-94 Response body parameters

Parameter	Type	Description
target_id	String	Target ID
creation_time	String	Time when the target was created
failure_details	FailureDetail sMessage object	Error information
file_system_path	String	Name of the interworking directory
lifecycle	String	Binding status. If the returned status is CREATING, you need to call the API for obtaining backend storage details to poll the binding completion status. If the returned status is AVAILABLE, the backend storage is successfully bound. If the status is MISCONFIGURED, the backend storage fails to be bound. The DELETING state is not supported currently.
obs	ObsDataRepository object	OBS target

Table 9-95 FailureDetailsMessage

Parameter	Type	Description
message	String	Error message

Table 9-96 ObsDataRepository

Parameter	Type	Description
bucket	String	OBS bucket name
endpoint	String	Name of the region where the bucket belongs
policy	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend
attributes	ObsTargetAttributes object	Properties of the storage backend. This parameter is not supported for file systems that are created on or before June 30, 2024 and have not been upgraded. Submit a service ticket if you need it.

Table 9-97 ObsDataRepositoryPolicy

Parameter	Type	Description
auto_export_policy	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-98 AutoExportPolicy

Parameter	Type	Description
events	Array of strings	Type of data automatically exported to the OBS bucket. <ul style="list-style-type: none">• NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket.• CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket.• DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Table 9-99 ObsTargetAttributes

Parameter	Type	Description
file_mode	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Type	Description
dir_mode	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	Integer	ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).
gid	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Status code: 400

Table 9-100 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-101 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

- This example adds a storage backend for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**. The OBS bucket name is **myBucket**, the OBS bucket endpoint is **obs.region.example.com**, and the name of the interworking directory is **sfsturboDirName**.

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/targets
```

```
{
  "file_system_path" : "sfsturboDirName",
  "obs" : {
    "bucket" : "myBucket",
    "endpoint" : "obs.region.example.com"
  }
}
```

- This example adds a storage backend for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**. The OBS bucket name is **myBucket**, the OBS bucket endpoint is **obs.region.example.com**, and the name of the interworking directory is **sfsturboDirName**. The permissions of imported files are set to **750**, and the permissions of imported directories are set to **640**.

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/targets
```

```
{
  "file_system_path" : "sfsturboDirName",
  "obs" : {
    "bucket" : "myBucket",
    "endpoint" : "obs.region.example.com",
    "attributes" : {
      "file_mode" : 750,
      "dir_mode" : 640
    }
  }
}
```

- This example adds a storage backend for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**. The OBS bucket name is **myBucket**, the OBS bucket endpoint is **obs.region.example.com**, and the name of the interworking directory is **sfsturboDirName**. The permissions of imported files are set to **750**, the permissions of imported directories are set to **640**, and both the UIDs and GIDs of the imported files and directories are set to **0**.

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/targets
```

```
{
  "file_system_path" : "sfsturboDirName",
  "obs" : {
    "bucket" : "myBucket",
    "endpoint" : "obs.region.example.com",
    "uid" : 0,
    "gid" : 0
  }
}
```

```
"attributes" : {  
  "file_mode" : 750,  
  "dir_mode" : 640,  
  "uid" : 0,  
  "gid" : 0  
}  
}
```

- This example adds a storage backend for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**. The OBS bucket name is **myBucket**, the OBS bucket endpoint is **obs.region.example.com**, and the name of the interworking directory is **sfsturboDirName**. The type of the auto export policy is set to **NEW**, **CHANGED**, and **DELETED**.

POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/targets

```
{  
  "file_system_path" : "sfsturboDirName",  
  "obs" : {  
    "bucket" : "myBucket",  
    "endpoint" : "obs.region.example.com",  
    "policy" : {  
      "auto_export_policy" : {  
        "events" : [ "NEW", "CHANGED", "DELETED" ]  
      }  
    }  
  }  
}
```

Example Responses

Status code: 202

Task submitted

```
{  
  "target_id" : "00000334-xxxx-402d-a5d4-bxxxxx87b939",  
  "creation_time" : "2023-11-19T04:02:03",  
  "file_system_path" : "sfsturboDirName",  
  "lifecycle" : "CREATING",  
  "obs" : {  
    "bucket" : "myBucket",  
    "endpoint" : "obs.region.example.com"  
  }  
}
```

Status code: 400

Error response

```
{  
  "errCode" : "SFS.TURBO.0001",  
  "errMsg" : "request path/body parameters invalid"  
}
```

Status code: 500

Error response

```
{  
  "errCode" : "SFS.TURBO.0005",  
  "errMsg" : "internal server error"  
}
```

Status Codes

Status Code	Description
202	Task submitted
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.2 Querying Backend Targets

Function

This API is used to query backend targets of a file system.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/targets

Table 9-102 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-103 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of targets returned in the list
marker	No	String	Query offset

Request Parameters

Table 9-104 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-105 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-106 Response body parameters

Parameter	Type	Description
count	Integer	Number of backend targets
targets	Array of ShowBackendTargetInfoResponseBody objects	List of backend targets

Table 9-107 ShowBackendTargetInfoResponseBody

Parameter	Type	Description
target_id	String	Target ID
creation_time	String	Time when the target was created
file_system_path	String	Name of the interworking directory
failure_details	FailureDetailsMessage object	Error information
lifecycle	String	Target status

Parameter	Type	Description
obs	ObsDataRepository object	OBS target

Table 9-108 FailureDetailsMessage

Parameter	Type	Description
message	String	Error message

Table 9-109 ObsDataRepository

Parameter	Type	Description
bucket	String	OBS bucket name
endpoint	String	Name of the region where the bucket belongs
policy	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend
attributes	ObsTargetAttributes object	Properties of the storage backend. This parameter is not supported for file systems that are created on or before June 30, 2024 and have not been upgraded. Submit a service ticket if you need it.

Table 9-110 ObsDataRepositoryPolicy

Parameter	Type	Description
auto_export_policy	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-111 AutoExportPolicy

Parameter	Type	Description
events	Array of strings	<p>Type of data automatically exported to the OBS bucket.</p> <ul style="list-style-type: none"> • NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Table 9-112 ObsTargetAttributes

Parameter	Type	Description
file_mode	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Type	Description
dir_mode	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	Integer	ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).
gid	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Status code: 404

Table 9-113 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-114 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Listing 10 storage backends starting from the one whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **630509b1-ded4-476e-8d06-dbbc3dc23900**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/630509b1-ded4-476e-8d06-dbbc3dc23900/targets?marker=11abef677ac40f46644d1d5cfc2424a4&limit=10
```

Example Responses

Status code: 200

Task submitted

```
{
  "count": 1,
  "targets": [ {
    "target_id": "00000334-xxxx-402d-a5d4-bxxxxx87b939",
    "creation_time": "2023-10-10T12:02:03",
    "file_system_path": "sfsturboDirName",
    "obs": {
      "bucket": "myBucket",
      "endpoint": "obs.region.example.com"
    }
  }
}]
```

Status code: 404

Error response

```
{
  "errCode": "SFS.TURBO.0002",
  "errMsg": "share not exist"
}
```

Status code: 500

Error response

```
{
  "errCode": "SFS.TURBO.0005",
  "errMsg": "Internal server error"
}
```

Status Codes

Status Code	Description
200	Task submitted

Status Code	Description
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.3 Obtaining Details About a Backend Target

Function

This API is used to obtain details about a backend target.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}

Table 9-115 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
target_id	Yes	String	Target ID

Request Parameters

Table 9-116 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-117 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-118 Response body parameters

Parameter	Type	Description
target_id	String	Target ID
creation_time	String	Time when the target was created
file_system_path	String	Name of the interworking directory
failure_details	FailureDetailsMessage object	Error information
lifecycle	String	Target status
obs	ObsDataRepository object	OBS target

Table 9-119 FailureDetailsMessage

Parameter	Type	Description
message	String	Error message

Table 9-120 ObsDataRepository

Parameter	Type	Description
bucket	String	OBS bucket name
endpoint	String	Name of the region where the bucket belongs
policy	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend

Parameter	Type	Description
attributes	ObsTargetAttributes object	Properties of the storage backend. This parameter is not supported for file systems that are created on or before June 30, 2024 and have not been upgraded. Submit a service ticket if you need it.

Table 9-121 ObsDataRepositoryPolicy

Parameter	Type	Description
auto_export_policy	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-122 AutoExportPolicy

Parameter	Type	Description
events	Array of strings	Type of data automatically exported to the OBS bucket. <ul style="list-style-type: none"> • NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Table 9-123 ObsTargetAttributes

Parameter	Type	Description
file_mode	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Type	Description
dir_mode	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	Integer	ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).
gid	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Status code: 404

Table 9-124 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-125 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Obtaining details of the storage backend whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/targets/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 200

Task submitted

```
{
  "target_id" : "00000334-xxxx-402d-a5d4-bxxxxx87b939",
  "creation_time" : "2023-10-10T12:02:03",
  "file_system_path" : "sfsturboDirName",
  "lifecycle" : "CREATING",
  "obs" : {
    "bucket" : "myBucket",
    "endpoint" : "obs.region.example.com"
  }
}
```

Status code: 404

Error response

```
{
  "errCode" : "SFS.TURBO.0126",
  "errMsg" : "target not found, not bound yet"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "internal server error"
}
```

Status Codes

Status Code	Description
200	Task submitted
404	Error response

Status Code	Description
500	Error response

Error Codes

See [Error Codes](#).

9.6.4 Deleting a Backend Target

Function

This API is used to delete a backend target.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}

Table 9-126 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
target_id	Yes	String	Target ID

Table 9-127 Query Parameters

Parameter	Mandatory	Type	Description
delete_data_in_file_system	No	Boolean	Whether to delete the corresponding file system interworking directory and data files in it. The default value is false . Deleted data cannot be recovered.

Request Parameters

Table 9-128 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 202

Table 9-129 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-130 Response body parameters

Parameter	Type	Description
target_id	String	Target ID
delete_data_in_file_system	Boolean	Whether to delete the corresponding file system interworking directory and data files in it.
lifecycle	String	Target status returned. Only DELETING and FAILED are supported.

Status code: 404

Table 9-131 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-132 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting the backend target whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/targets/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 202

Task submitted

```
{
  "target_id" : "00000334-xxxx-402d-a5d4-bxxxxx87b939",
  "delete_data_in_file_system" : false,
  "life_cycle" : "DELETING"
}
```

Status code: 404

Error response

```
{
  "errCode" : "SFS.TURBO.0126",
  "errMsg" : "target not found, not bound yet"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "internal server error"
}
```

Status Codes

Status Code	Description
202	Task submitted
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.5 Updating the Properties of a Storage Backend

Function

This API is used to update the properties of a storage backend.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}/attributes

Table 9-133 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
target_id	Yes	String	ID of the storage backend

Request Parameters

Table 9-134 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-135 Request body parameters

Parameter	Mandatory	Type	Description
attributes	Yes	ObsTargetAttributes object	Properties of the storage backend

Table 9-136 ObsTargetAttributes

Parameter	Mandatory	Type	Description
file_mode	No	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Mandatory	Type	Description
dir_mode	No	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	No	Integer	<p>ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).</p>

Parameter	Mandatory	Type	Description
gid	No	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Response Parameters

Status code: 200

Table 9-137 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-138 Response body parameters

Parameter	Type	Description
target_id	String	ID of the storage backend
attributes	ObsTargetAttributes object	Properties of the storage backend

Table 9-139 ObsTargetAttributes

Parameter	Type	Description
file_mode	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Type	Description
dir_mode	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	Integer	ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).
gid	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Status code: 404

Table 9-140 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-141 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Updating the properties of the storage backend whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
PUT HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/targets/11abef677ac40f46644d1d5cfc2424a4/attributes
```

```
{
  "attributes": {
    "file_mode": 750,
    "dir_mode": 640,
    "uid": 0,
    "gid": 0
  }
}
```

Example Responses

Status code: 200

Success

```
{
  "target_id": "00000334-xxxx-402d-a5d4-bxxxxx87b939",
  "attributes": {
    "file_mode": 750,
    "dir_mode": 640,
    "uid": 0,
    "gid": 0
  }
}
```

Status code: 404

Error response

```
{
  "errCode": "SFS.TURBO.0126",
  "errMsg": "target not found, not bound yet"
}
```

Status code: 500

Error response

```
{
  "errCode": "SFS.TURBO.0005",
  "errMsg": "internal server error"
}
```

Status Codes

Status Code	Description
200	Success
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.6 Updating the Auto Synchronization Policy of a Storage Backend

Function

This API is used to update the auto synchronization policy of a storage backend.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}/policy

Table 9-142 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
target_id	Yes	String	Storage backend ID

Request Parameters

Table 9-143 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	MIME type

Table 9-144 Request body parameters

Parameter	Mandatory	Type	Description
policy	Yes	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend

Table 9-145 ObsDataRepositoryPolicy

Parameter	Mandatory	Type	Description
auto_export_policy	No	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-146 AutoExportPolicy

Parameter	Mandatory	Type	Description
events	No	Array of strings	<p>Type of data automatically exported to the OBS bucket.</p> <ul style="list-style-type: none"> • NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Response Parameters

Status code: 200

Table 9-147 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-148 Response body parameters

Parameter	Type	Description
target_id	String	Storage backend ID

Parameter	Type	Description
policy	ObsDataRepositoryPolicy object	Auto synchronization policy of the storage backend

Table 9-149 ObsDataRepositoryPolicy

Parameter	Type	Description
auto_export_policy	AutoExportPolicy object	Auto export policy of the storage backend. If enabled, all updates made on the file system will be automatically exported to the OBS bucket.

Table 9-150 AutoExportPolicy

Parameter	Type	Description
events	Array of strings	Type of data automatically exported to the OBS bucket. <ul style="list-style-type: none"> • NEW: Files created and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • CHANGED: Files previously imported from the OBS bucket and then modified in the SFS Turbo interworking directory. Any data or metadata modifications made will be automatically synchronized to the OBS bucket. • DELETED: Files deleted from the SFS Turbo interworking directory. Deletions will be automatically synchronized to the OBS bucket, and only such files that were previously exported to the bucket will be deleted.

Status code: 404

Table 9-151 Response body parameters

Parameter	Type	Description
errCode	String	Error code

Parameter	Type	Description
errMsg	String	Error description

Status code: 500

Table 9-152 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Updating the auto synchronization policy of the storage backend whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
PUT HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/targets/11abef677ac40f46644d1d5cfc2424a4/policy
```

```
{
  "policy": {
    "auto_export_policy": {
      "events": [ "NEW", "CHANGED", "DELETED" ]
    }
  }
}
```

Example Responses

Status code: 200

Successful

```
{
  "target_id": "xxxxxx-xxx-xxxx-xxxxxxx",
  "policy": {
    "auto_export_policy": {
      "events": [ "NEW", "CHANGED", "DELETED" ]
    }
  }
}
```

Status code: 404

Error response

```
{
  "errCode": "SFS.TURBO.0126",
  "errMsg": "target not found, not bound yet"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "internal server error"
}
```

Status Codes

Status Code	Description
200	Successful
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.7 Creating an Import or Export Task

Function

This API is used to create an import or export task.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

POST /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task

Table 9-153 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-154 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	MIME type

Table 9-155 Request body parameters

Parameter	Mandatory	Type	Description
type	Yes	String	<p>Task type, which can be import (additional metadata import), import_metadata (quick import), preload (data preload), or export (export).</p> <p>import: Object metadata, including the name, size, and last modified time, as well as the additional metadata like uid, gid, and mode previously exported from SFS Turbo will all be imported.</p> <p>import_metadata: Only the object metadata, including the name, size, and last modified time will be imported. After the import, SFS Turbo will, by default, generate the additional metadata.</p> <p>preload: Both the metadata and data will be imported. The metadata includes only the object metadata. Additional metadata like uid, gid, and mode will not be imported.</p> <p>export: SFS Turbo will export to the OBS bucket the files created in the interworking directory as well as the data previously imported from OBS and then modified in SFS Turbo.</p>
src_target	Yes	String	Name of the interworking directory

Parameter	Mandatory	Type	Description
src_prefix	No	String	Prefix of the source path of an import or export task. The OBS bucket name does not need to be included during import, and the linkage directory name does not need to be included during export. For data preheating import, the source path prefix must be a directory or an object that ends with a slash (/). If this field is not specified, all objects in the bound OBS bucket are imported, and all files in the linkage directory are exported.
dest_target	Yes	String	Keep the same as src_target .
dest_prefix	No	String	Keep the same as src_prefix .
attributes	No	ObsTargetAttributes object	Properties of the storage backend

Table 9-156 ObsTargetAttributes

Parameter	Mandatory	Type	Description
file_mode	No	Integer	<p>Permissions on the imported file. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the file owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the file belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The file owner is specified by UID, and the user group to which the file belongs is specified by GID. Users who are not the file owner and not in the user group to which the file belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the file owner has the read, write, and execute permissions on the file, the second digit 5 indicates that the user group to which the file belongs has the read and execute permissions on the file, and the third digit 0 indicates that other users have no permission on the file.</p>

Parameter	Mandatory	Type	Description
dir_mode	No	Integer	<p>Permissions on the imported directory. Value range: 0 to 777</p> <p>The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7. The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7. The third digit indicates the permissions of other users, and its value ranges from 0 to 7. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users.</p> <p>Values 4, 2, and 1 indicate the read, write, and execute permissions respectively. The total value between 1 and 7 represents the access permissions. For example, the first digit 7 in 750 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 0 indicates that other users have no permission on the directory.</p>
uid	No	Integer	<p>ID of the user who imports the object. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).</p>

Parameter	Mandatory	Type	Description
gid	No	Integer	ID of the user group to which the imported object belongs. The default value is 0. The value ranges from 0 to 4,294,967,294 ($2^{32} - 2$).

Response Parameters

Status code: 202

Table 9-157 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-158 Response body parameters

Parameter	Type	Description
task_id	String	Task ID

Status code: 400

Table 9-159 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-160 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-161 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-162 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

- Creating an import task and choosing to import the metadata (with the interworking directory name set to **sfs-link-directory** and prefix of the source path in the OBS bucket set to **input/datasets/**)

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task
```

```
{
  "type" : "import_metadata",
  "src_target" : "sfs-link-directory",
  "src_prefix" : "input/datasets/",
  "dest_target" : "sfs-link-directory",
  "dest_prefix" : "input/datasets/"
}
```

- Creating an import task and choosing to import the metadata (with the interworking directory name set to **sfs-link-directory**, the prefix of the source path in the OBS bucket set to **input/datasets/**, the permissions of the imported files set to **755**, and the permissions of the imported directories set to **755**)

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task
```

```
{
  "type" : "import_metadata",
  "src_target" : "sfs-link-directory",
  "src_prefix" : "input/datasets/",
  "dest_target" : "sfs-link-directory",
  "dest_prefix" : "input/datasets/",
  "attributes" : {
    "file_mode" : 755,
    "dir_mode" : 755
  }
}
```

- Creating an import task and choosing to preload data (with the interworking directory name set to **sfs-link-directory**, the prefix of the source path in the OBS bucket set to **input/datasets/**, the permissions of the imported files set to **755**, the permissions of the imported directories set to **755**, and the UID and GID both set to **0**)

```
POST HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task
```

```
{
  "type" : "preload",
  "src_target" : "sfs-link-directory",
}
```

```

"src_prefix" : "input/datasets/",
"dest_target" : "sfs-link-directory",
"dest_prefix" : "input/datasets/",
"attributes" : {
  "file_mode" : 755,
  "dir_mode" : 755,
  "uid" : 0,
  "gid" : 0
}
}

```

Example Responses

Status code: 202

Accepted

```

{
  "task_id" : "7bd2a9b6-xxxx-4605-xxxx-512d636001b0"
}

```

Status code: 400

Client error

```

{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "parameter error"
}

```

Status code: 500

Internal error

```

{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}

```

Status Codes

Status Code	Description
202	Accepted
400	Client error
500	Internal error

Error Codes

See [Error Codes](#).

9.6.8 Querying Details About an Import or Export Task

Function

This API is used to query details about an import or export task.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

GET /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task/{task_id}

Table 9-163 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
task_id	Yes	String	Task ID

Request Parameters

Table 9-164 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-165 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-166 Response body parameters

Parameter	Type	Description
task_id	String	Task ID
type	String	Task type
status	String	Task status

Parameter	Type	Description
src_target	String	Name of the interworking directory
src_prefix	String	Source path prefix specified in an import or export task
dest_target	String	Keep it the same as src_target .
dest_prefix	String	Keep it the same as src_prefix .
start_time	String	Task start time
end_time	String	Task end time
message	String	Task execution results

Status code: 400

Table 9-167 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-168 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-169 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-170 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Query details about a task, with the file system ID **77ba6f4b-6365-4895-8dda-bc7142af4dde** and task ID **11abef677ac40f46644d1d5cfc2424a4**

```
"GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/77ba6f4b-6365-4895-8dda-bc7142af4dde/hpc-cache/task/11abef677ac40f46644d1d5cfc2424a4"
```

Example Responses

Status code: 200

Success

```
{
  "task_id": "7bd2a9b6-xxxx-4605-xxxx-512d636001b0",
  "type": "import_metadata",
  "status": "DOING",
  "src_target": "sfs-link-directory",
  "src_prefix": "input/datasets/",
  "dest_target": "sfs-link-directory",
  "dest_prefix": "input/datasets/",
  "message": "",
  "start_time": "2023-09-02T15:04:05",
  "end_time": ""
}
```

Status code: 400

Error response

```
{
  "errCode": "SFS.TURBO.0001",
  "errMsg": "parameter error"
}
```

Status code: 500

Error response

```
{
  "errCode": "SFS.TURBO.0005",
  "errMsg": "Internal server error"
}
```

Status Codes

Status Code	Description
200	Success
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.9 Listing Import and Export Tasks

Function

This API is used to list import and export tasks.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

GET /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/tasks

Table 9-171 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-172 Query Parameters

Parameter	Mandatory	Type	Description
type	No	String	Task type
status	No	String	Task status
offset	No	Long	Offset. The default value is 0 .
limit	No	Long	Limit. The default value is 20 .
start_time	No	String	start_time
end_time	No	String	end_time

Request Parameters

Table 9-173 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-174 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-175 Response body parameters

Parameter	Type	Description
tasks	Array of OneHpcCacheTaskInfoResp objects	Task details
count	Long	Number of tasks

Table 9-176 OneHpcCacheTaskInfoResp

Parameter	Type	Description
task_id	String	Task ID
type	String	Task type
status	String	Task status
src_target	String	Name of the interworking directory
src_prefix	String	Source path prefix specified in an import or export task
dest_target	String	Keep it the same as src_target .
dest_prefix	String	Keep it the same as src_prefix .
start_time	String	Task start time
end_time	String	Task end time
message	String	Task execution results

Status code: 400

Table 9-177 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-178 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-179 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-180 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Listing import and export tasks for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET /v1/{project_id}/sfs-turbo/77ba6f4b-6365-4895-8dda-bc7142af4dde/hpc-cache/tasks
```

Example Responses

Status code: 200

Success

```
{
  "tasks": [ {
    "task_id": "7bd2a9b6-xxxx-4605-xxxx-512d636001b0",
    "type": "import",
    "status": "DOING",
    "src_target": "sfs-link-directory",
    "src_prefix": "input/datasets/",
    "dest_target": "sfs-link-directory",
```

```

"dest_prefix" : "input/datasets/",
"message" : "",
"start_time" : "2023-09-02T15:04:05",
"end_time" : ""
}],
"count" : 1
}

```

Status code: 400

Error response

```

{
"errCode" : "SFS.TURBO.0001",
"errMsg" : "parameter error"
}

```

Status code: 500

Error response

```

{
"errCode" : "SFS.TURBO.0005",
"errMsg" : "Internal server error"
}

```

Status Codes

Status Code	Description
200	Success
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.10 Deleting an Import or Export Task

Function

This API is used to delete an import or export task.

Constraints

This API is only supported for SFS Turbo 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

URI

DELETE /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task/{task_id}

Table 9-181 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
task_id	Yes	String	Task ID

Request Parameters

Table 9-182 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 202

Table 9-183 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Status code: 400

Table 9-184 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-185 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-186 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-187 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting a task whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
"DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/77ba6f4b-6365-4895-8dda-bc7142af4dde/hpc-cache/task/11abef677ac40f46644d1d5cfc2424a4"
```

Example Responses

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "parameter error"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
202	Success
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.6.11 Updating a File System

Function

This API is used to update the cold data eviction period of a file system.

Constraints

This API is only supported for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems that already have storage backends added.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}

Table 9-188 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-189 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-190 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Type of operation for updating a file system Only config_gc_time is supported currently.

Parameter	Mandatory	Type	Description
gc_time	Yes	Integer	Cold data eviction duration. The unit is hour. The value ranges from 1 to 100000000 . The default value is 60 hours.

Response Parameters

Status code: 200

Table 9-191 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-192 Response body parameters

Parameter	Type	Description
gc_time	Integer	File system cold data eviction duration

Status code: 404

Table 9-193 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-194 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

```
{  
  "action": "config_gc_time",  
  "gc_time": 5  
}
```

Example Responses

Status code: 200

Task submitted

```
{  
  "gc_time": 10  
}
```

Status code: 404

Error response

```
{  
  "errCode": "SFS.TURBO.0002",  
  "errMsg": "share not exist"  
}
```

Status code: 500

Error response

```
{  
  "errCode": "SFS.TURBO.0005",  
  "errMsg": "Internal server error"  
}
```

Status Codes

Status Code	Description
200	Task submitted
404	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.7 Directory Management

9.7.1 Creating Quotas for a Directory

Function

This API is used to create quotas for a directory.

Constraints

You can only configure directory quotas on empty directories.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-quota

Table 9-195 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-196 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-197 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory
capacity	No	Integer	Size of the directory, in MB. If it is set to 0 , data cannot be written to the directory. Use either capacity or quota .
inode	No	Integer	Quantity limit of inodes. If it is set to 0 , data cannot be written to the directory. Use either capacity or quota .

Response Parameters

Status code: 200

Table 9-198 Response body parameters

Parameter	Type	Description
path	String	Valid full path of an existing directory
capacity	Integer	Size of the directory, in MB
inode	Integer	Maximum number of inodes allowed in the directory
used_capacity	Integer	Used space of the directory, in MB. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.
used_inode	Integer	Number of inodes used in the directory. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

Status code: 400

Table 9-199 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 409

Table 9-200 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-201 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Creating quotas for the `/data/test` directory, with the capacity quota set to **1024** MB and number of inodes set to **100000**.

```
{
  "path" : "/data/test",
  "capacity" : 1024,
  "inode" : 100000
}
```

Example Responses

Status code: 200

Directory quotas created

```
{
  "path" : "/data/test",
  "capacity" : 1024,
  "inode" : 100000
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0102",
  "errMsg" : "Path is not directory"
}
```

Status code: 409

Error response

```
{
  "errCode" : "SFS.TURBO.0112",
  "errMsg" : "quota already exist"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Directory quotas created
400	Error response
409	Error response

Status Code	Description
500	Error response

Error Codes

See [Error Codes](#).

9.7.2 Updating Quotas of a Directory

Function

This API is used to update quotas of a directory.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-quota

Table 9-202 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-203 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-204 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory

Parameter	Mandatory	Type	Description
capacity	No	Integer	Size of the directory, in MB. If it is set to 0 , data cannot be written to the directory. Use either capacity or quota .
inode	No	Integer	Quantity limit of inodes. If it is set to 0 , data cannot be written to the directory. Use either capacity or quota .

Response Parameters

Status code: 200

Table 9-205 Response body parameters

Parameter	Type	Description
path	String	Valid full path of an existing directory
capacity	Integer	Size of the directory, in MB
inode	Integer	Maximum number of inodes allowed in the directory
used_capacity	Integer	Used space of the directory, in MB. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.
used_inode	Integer	Number of inodes used in the directory. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

Status code: 400

Table 9-206 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 403

Table 9-207 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-208 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Updating quotas of the **/data/test** directory, with the capacity quota set to **1024** MB and number of inodes set to **100000**.

```
{
  "path" : "/data/test",
  "capacity" : 1024,
  "inode" : 100000
}
```

Example Responses

Status code: 200

Directory quotas updated

```
{
  "path" : "/data/test",
  "capacity" : 1024,
  "inode" : 100000
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0102",
  "errMsg" : "Path is not directory"
}
```

Status code: 403

Error response

```
{
  "errCode" : "SFS.TURBO.0113",
  "errMsg" : "dir not create quota"
}
```


Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Directory quotas updated
400	Error response
403	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.7.3 Querying Quotas of a Directory

Function

This API is used to update quotas of a directory. The **used_capacity** and **used_inode** obtained may not be the latest.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-quota

Table 9-209 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-210 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory

Request Parameters

Table 9-211 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-212 Response body parameters

Parameter	Type	Description
path	String	Valid full path of an existing directory
capacity	Integer	Size of the directory, in MB
inode	Integer	Maximum number of inodes allowed in the directory
used_capacity	Integer	Used space of the directory, in MB. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems. The returned data may not be the latest.
used_inode	Integer	Number of inodes used in the directory. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems. The returned data may not be the latest.

Status code: 400

Table 9-213 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-214 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Querying the quotas of the directory **/data/test** in the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4ddw**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4ddw/fs/dir-quota?path=/data/test
```

Example Responses

Status code: 200

Directory quotas queried

```
{
  "path" : "/data/test",
  "capacity" : 1024,
  "inode" : 100000
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0102",
  "errMsg" : "Path is not directory"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Directory quotas queried
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.7.4 Deleting Quotas of a Directory

Function

This API is used to delete quotas of a directory.

Constraints

Only empty directories can have their quotas deleted.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-quota

Table 9-215 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-216 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-217 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory

Response Parameters

Status code: 400

Table 9-218 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-219 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting the quotas of the directory **/data/test** in the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4ddw**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4ddw/fs/
dir-quota
{
  "path" : "/data/test"
}
```

Example Responses

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0102",
  "errMsg" : "Path is not directory"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
204	Directory quotas deleted
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.7.5 Creating a Directory

Function

This API is used to create a directory.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir

Table 9-220 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-221 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-222 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory
mode	No	Long	Directory permissions, which range from 0 to 777* . The default value is 755 . The first digit indicates the permissions of the directory owner, and its value ranges from 0 to 7 . The second digit indicates the permissions of the user group to which the directory belongs, and its value ranges from 0 to 7 . The third digit indicates the permissions of other users, and its value ranges from 0 to 7 . The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users. For example, in 755, the first digit 7 indicates that the directory owner has the read, write, and execute permissions on the directory, the second digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the third digit 5 indicates that other users have the read and execute permissions on the directory.
uid	No	Long	ID of the directory owner. The default value is 0 . The value ranges from 0 to 4,294,967,294 ($2^{32}-2$).
gid	No	Long	ID of the user group to which the directory belongs. The default value is 0 . The value ranges from 0 to 4,294,967,294 ($2^{32}-2$).

Response Parameters

Status code: 400

Table 9-223 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 409

Table 9-224 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-225 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Creating a directory whose full path is **/date/test**

```
{
  "path" : "/date/test"
}
```

Example Responses

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0122",
  "errMsg" : "invalid mode"
}
```

Status code: 409

Conflict directory

```
{
  "errCode" : "SFS.TURBO.0114",
  "errMsg" : "path already exist"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
204	Directory created
400	Error response
409	Conflict directory
500	Error response

Error Codes

See [Error Codes](#).

9.7.6 Checking Whether a Directory Exists

Function

This API is used to query whether a directory exists.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir

Table 9-226 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-227 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Full path of the directory to be queried

Request Parameters

Table 9-228 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-229 Response body parameters

Parameter	Type	Description
path	String	Full path of the directory
mode	Long	Permissions of the directory. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems. The third digit indicates the permissions of the directory owner, the fourth digit indicates the permissions of the user group to which the directory belongs, and the fifth digit indicates the permissions of other users. The directory owner is specified by UID, and the user group to which the directory belongs is specified by GID. Users who are not the directory owner and not in the user group to which the directory belongs are other users. For example, in 40755, the third digit 7 indicates that the directory owner has the read, write, and execute permissions on the directory, the fourth digit 5 indicates that the user group to which the directory belongs has the read and execute permissions on the directory, and the fifth digit 5 indicates that other users have the read and execute permissions on the directory.

Parameter	Type	Description
uid	Long	ID of the user who owns the directory. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.
gid	Long	ID of the user group to which the directory belongs. This field is returned only for 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems.

Status code: 400

Table 9-230 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 404

Table 9-231 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-232 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Querying whether the directory **/date/test** can be found in the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir?path=/date/test
```

Example Responses

Status code: 200

Query results

```
{  
  "path" : "/date/test"  
}
```

Status code: 400

Error response

```
{  
  "errCode" : "SFS.TURBO.0100",  
  "errMsg" : "invalid path"  
}
```

Status code: 404

Directory not found

```
{  
  "errCode" : "SFS.TURBO.0101",  
  "errMsg" : "path not exist"  
}
```

Status code: 500

Error response

```
{  
  "errCode" : "SFS.TURBO.0005",  
  "errMsg" : "Internal server error"  
}
```

Status Codes

Status Code	Description
200	Query results
400	Error response
404	Directory not found
500	Error response

Error Codes

See [Error Codes](#).

9.7.7 Deleting a Directory

Function

This API is used to delete a directory.

Constraints

This API is only supported for file systems created after June 1, 2023.

Deleting a directory from a file system is a risky operation. Once deleted, the directory cannot be recovered. Ensure that the directory you specify is the one you want to delete.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir

Table 9-233 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-234 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-235 Request body parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of an existing directory

Response Parameters

Status code: 400

Table 9-236 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-237 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting the `/test` directory

```
{
  "path" : "/test"
}
```

Example Responses

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
202	Deletion request accepted
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.7.8 Querying the Usage of a File System Directory

Function

This API is used to query the usage of a file system directory (including usages of subdirectories). The obtained data may not be the latest as there is a 5-minute delay between the frontend and backend.

Constraints

This API is only supported for file systems created after August 1, 2023. This API is only supported for previous-generation file system types (Standard, Standard-Enhanced, Performance, and Performance-Enhanced). For 1,000 MB/s/TiB, 500 MB/s/TiB, 250 MB/s/TiB, 125 MB/s/TiB, 40 MB/s/TiB, and 20 MB/s/TiB file systems, use the API for creating asynchronous tasks. You are advised not to submit five or more requests at a time, or the file system performance may be affected. It takes a long time to query an oversized directory. Do not submit the request repeatedly.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/dir-usage

Table 9-238 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-239 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Valid full path of a directory in the file system

Request Parameters

Table 9-240 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type, which can be application or json

Response Parameters

Status code: 200

Table 9-241 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-242 Response body parameters

Parameter	Type	Description
dir_usage	FsDirUasge object	Message description

Table 9-243 FsDirUasge

Parameter	Type	Description
used_capacity	Long	Used capacity, in byte

Status code: 400

Table 9-244 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-245 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-246 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-247 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Querying the directory usage of a Standard file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/dir-usage
```

Example Responses

Status code: 200

Directory resource usages

```
{
  "dir_usage" : {
    "used_capacity" : 1024000
  }
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0102",
  "errMsg" : "Path is not directory"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Directory resource usages
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8 Permissions Management

9.8.1 Creating a Permission Rule

Function

This API is used to create a permission rule.

Constraints

A maximum of 64 permissions rules can be configured for a file system.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/perm-rules

Table 9-248 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-249 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-250 Request body parameters

Parameter	Mandatory	Type	Description
rules	Yes	Array of OnePermRuleRequestInfo objects	Permission rule details. A maximum of five rules can be created at a time.

Table 9-251 OnePermRuleRequestInfo

Parameter	Mandatory	Type	Description
ip_cidr	No	String	IP address or IP address range of the object to be authorized. Once configured, this parameter cannot be modified.
rw_type	No	String	Read/write permission of the object to be authorized. <ul style="list-style-type: none"> • rw: read and write permission, which is the default option • ro: read-only permission • none: no permission

Parameter	Mandatory	Type	Description
user_type	No	String	<p>System user's permission to access the file system. The value can be any of the following:</p> <ul style="list-style-type: none"> no_root_squash: default option. The client uses any user, including the root user. The NFS server retains the user used by the client and does not map the user. root_squash: When the client uses the root user, the user mapped to the NFS server is the NFS anonymous user (nfsnobody). If the client uses a non-root user, the NFS server retains the user used by the client and does not map the user. all_squash: All users of clients that access the NFS server are mapped as anonymous users.

Response Parameters

Status code: 200

Table 9-252 Response body parameters

Parameter	Type	Description
rules	Array of OnePermRuleResponseInfo objects	Permission rule details

Table 9-253 OnePermRuleResponseInfo

Parameter	Type	Description
id	String	Permission rule ID
ip_cidr	String	IP address or IP address range of the authorized object

Parameter	Type	Description
rw_type	String	Read/write permission of the authorized object. <ul style="list-style-type: none"> • rw: read and write permission, which is the default option • ro: read-only permission • none: no permission
user_type	String	File system access permission granted to the user of the authorized object. Supported values are: <ul style="list-style-type: none"> • no_root_squash: allows the root user on the client to access the file system as root. • root_squash: allows the root user on the client to access the file system as nfsnobody. • all_squash: allows any user on the client to access the file system as nfsnobody. It is the default value.

Status code: 400

Table 9-254 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-255 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

```
{
  "rules": [ {
    "ip_cidr": "192.168.xx.xx/16",
    "rw_type": "rw",
    "user_type": "no_root_squash"
  }
]
```

```
}, {  
  "ip_cidr": "192.32.xx.xx/16",  
  "rw_type": "rw",  
  "user_type": "no_root_squash"  
}]  
}
```

Example Responses

Status code: 200

Successful creation

```
{  
  "rules": [ {  
    "id": "1131ed520xxxxxbedb6e57xxxxxxx",  
    "ip_cidr": "192.32.0.0/16",  
    "rw_type": "rw",  
    "user_type": "no_root_squash"  
  }, {  
    "id": "1131ed520xxxxxbedb6e57xxxxxxx",  
    "ip_cidr": "192.32.0.1",  
    "rw_type": "rw",  
    "user_type": "no_root_squash"  
  } ]  
}
```

Status code: 400

Error response

```
{  
  "errCode": "SFS.TURBO.0001",  
  "errMsg": "Rules not allowed empty"  
}
```

Status code: 500

Error response

```
{  
  "errCode": "SFS.TURBO.0005",  
  "errMsg": "Internal server error"  
}
```

Status Codes

Status Code	Description
200	Successful creation
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8.2 Querying Permission Rules of a File System

Function

This API is used to query the permission rules of a file system.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/perm-rules

Table 9-256 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Table 9-257 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Long	Number of returned permission rules.
offset	No	Long	Offset of the returned permission rule.

Request Parameters

Table 9-258 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-259 Response body parameters

Parameter	Type	Description
rules	Array of OnePermRuleResponseInfo objects	Permission rule information

Table 9-260 OnePermRuleResponseInfo

Parameter	Type	Description
id	String	Permission rule ID
ip_cidr	String	IP address or IP address range of the authorized object
rw_type	String	Read/write permission of the authorized object. <ul style="list-style-type: none"> • rw: read and write permission, which is the default option • ro: read-only permission • none: no permission
user_type	String	File system access permission granted to the user of the authorized object. Supported values are: <ul style="list-style-type: none"> • no_root_squash: allows the root user on the client to access the file system as root. • root_squash: allows the root user on the client to access the file system as nfsnobody. • all_squash: allows any user on the client to access the file system as nfsnobody. It is the default value.

Status code: 500

Table 9-261 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Querying the permission rules of the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/perm-rules
```

Example Responses

Status code: 200

Successful query

```
{
  "rules": [ {
    "id": "1131ed520xxxxxbedb6e57xxxxxxx",
    "ip_cidr": "192.168.xx.xx/16",
    "rw_type": "rw",
    "user_type": "no_root_squash"
  }, {
    "id": "1231ed520xxxxxbedb6e57xxxxxxx",
    "ip_cidr": "192.32.xx.xx/16",
    "rw_type": "rw",
    "user_type": "no_root_squash"
  } ]
}
```

Status code: 500

Error response

```
{
  "errCode": "SFS.TURBO.0005",
  "errMsg": "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful query
500	Error response

Error Codes

See [Error Codes](#).

9.8.3 Querying a Permission Rule of a File System

Function

This API is used to query a specific permission rule of a file system.

URI

```
GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/perm-rules/{rule_id}
```

Table 9-262 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
rule_id	Yes	String	Permission rule ID

Request Parameters

Table 9-263 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-264 Response body parameters

Parameter	Type	Description
id	String	Permission rule ID
ip_cidr	String	IP address or IP address range of the authorized object
rw_type	String	Read/write permission of the authorized object. <ul style="list-style-type: none">• rw: read and write permission, which is the default option• ro: read-only permission• none: no permission

Parameter	Type	Description
user_type	String	File system access permission granted to the user of the authorized object. Supported values are: <ul style="list-style-type: none"> • no_root_squash: allows the root user on the client to access the file system as root. • root_squash: allows the root user on the client to access the file system as nfsnobody. • all_squash: allows any user on the client to access the file system as nfsnobody. It is the default value.

Status code: 400

Table 9-265 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-266 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Querying details about the permission rule whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
GET HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/perm-rules/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 200

Successful query

```
{
  "id" : "1131ed520xxxxxxeb6e57xxxxxxx",
  "ip_cidr" : "192.168.xx.xx/16",
  "rw_type" : "rw",
  "user_type" : "no_root_squash"
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "Invalid rule id"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful query
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8.4 Modifying a Permission Rule

Function

This API is used to modify a permission rule.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/perm-rules/{rule_id}

Table 9-267 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Parameter	Mandatory	Type	Description
rule_id	Yes	String	Permission rule ID

Request Parameters

Table 9-268 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-269 Request body parameters

Parameter	Mandatory	Type	Description
ip_cidr	No	String	IP address or IP address range of the object to be authorized. Once configured, this parameter cannot be modified.
rw_type	No	String	Read/write permission of the object to be authorized. <ul style="list-style-type: none">• rw: read and write permission, which is the default option• ro: read-only permission• none: no permission

Parameter	Mandatory	Type	Description
user_type	No	String	<p>System user's permission to access the file system. The value can be any of the following:</p> <ul style="list-style-type: none"> • no_root_squash: default option. The client uses any user, including the root user. The NFS server retains the user used by the client and does not map the user. • root_squash: When the client uses the root user, the user mapped to the NFS server is the NFS anonymous user (nfsnobody). If the client uses a non-root user, the NFS server retains the user used by the client and does not map the user. • all_squash: All users of clients that access the NFS server are mapped as anonymous users.

Response Parameters

Status code: 200

Table 9-270 Response body parameters

Parameter	Type	Description
id	String	Permission rule ID
ip_cidr	String	IP address or IP address range of the authorized object
rw_type	String	<p>Read/write permission of the authorized object.</p> <ul style="list-style-type: none"> • rw: read and write permission, which is the default option • ro: read-only permission • none: no permission

Parameter	Type	Description
user_type	String	File system access permission granted to the user of the authorized object. Supported values are: <ul style="list-style-type: none"> • no_root_squash: allows the root user on the client to access the file system as root. • root_squash: allows the root user on the client to access the file system as nfsnobody. • all_squash: allows any user on the client to access the file system as nfsnobody. It is the default value.

Status code: 400

Table 9-271 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-272 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

```
{
  "rw_type": "rw",
  "user_type": "no_root_squash"
}
```

Example Responses

Status code: 200

Successful creation

```
{
  "id": "1131ed520xxxxxxebdb6e57xxxxxxx",
  "ip_cidr": "192.32.0.0/16",
}
```

```
"rw_type" : "rw",
"user_type" : "no_root_squash"
}
```

Status code: 400

Error response

```
{
"errCode" : "SFS.TURBO.0001",
"errMsg" : "Invalid rule id"
}
```

Status code: 500

Error response

```
{
"errCode" : "SFS.TURBO.0005",
"errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful creation
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8.5 Deleting a Permissions Rule

Function

This API is used to delete a permission rule.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/perm-rules/{rule_id}

Table 9-273 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID
rule_id	Yes	String	Permission rule ID

Request Parameters

Table 9-274 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 400

Table 9-275 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-276 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Deleting the permission rule whose ID is **11abef677ac40f46644d1d5cfc2424a4** for the file system whose ID is **77ba6f4b-6365-4895-8dda-bc7142af4dde**

```
DELETE HTTPS://{endpoint}/v1/{project_id}/sfs-turbo/shares/77ba6f4b-6365-4895-8dda-bc7142af4dde/fs/perm-rules/11abef677ac40f46644d1d5cfc2424a4
```

Example Responses

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
204	Successful deletion
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8.6 Creating and Binding the LDAP Configuration

Function

This API is used to create and bind the LDAP configuration. Lightweight Directory Access Protocol (LDAP) is a standard protocol used for accessing and controlling directory servers. An LDAP server can centrally manage the relationship between users and groups. After an LDAP server is bound, when a user accesses a file in your file system, SFS Turbo accesses your LDAP server for user authentication and obtains the relationship between users and groups. In this way, standards Linux file UGO permissions are checked. To use this function, you need to first set up an LDAP server. SFS Turbo only supports LDAP v3 currently. Common directory servers that provide LDAP access include OpenLDAP (Linux) and Active Directory (Windows). The implementation varies depending on the directory server. When binding an LDAP server, you need to specify the corresponding schema. If the configured schema is incorrect, SFS Turbo cannot obtain the correct user and group information. As a result, users may fail to access files in the file system. Schemas that SFS Turbo supports include:

1. RFC2307 (Usually selected for OpenLDAP)
2. MS-AD-BIS (Usually selected for Active Directory. It supports RFC2307bis and nested groups.)

SFS Turbo also supports active and standby LDAP servers. If one LDAP server fails and cannot be accessed, SFS Turbo automatically switches to the standby LDAP server. In addition, if you set **allow_local_user** to **Yes** (default value is **No**), when both LDAP servers fail, SFS Turbo will use your local user and group information instead of the information configured on the LDAP server for identity authentication and UGO permissions check to minimize the fault impact.

Constraints

base_dn and **url** cannot be empty.

URI

POST /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/ldap

Table 9-277 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-278 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-279 Request body parameters

Parameter	Mandatory	Type	Description
url	Yes	String	URL of the LDAP server. The format is <i>ldap://{ip_address}:{port_number}</i> or <i>ldaps://{ip_address}:{port_number}</i> , for example, ldap://192.168.xx.xx:60000 .
base_dn	Yes	String	Base DN
user_dn	No	String	User DN
password	No	String	LDAP authentication password
vpc_id	No	String	ID of the VPC which the specified LDAP server can be connected to. This parameter is required only when the SFS Turbo file system is used across VPCs.
filter_condition	No	String	Filter criteria. This is a reserved field and is not supported currently.
backup_url	No	String	URL of the standby LDAP server. The format is <i>ldap://{ip_address}:{port_number}</i> or <i>ldaps://{ip_address}:{port_number}</i> , for example, ldap://192.168.xx.xx:60000 .

Parameter	Mandatory	Type	Description
schema	No	String	LDAP schema. If it is not specified, RFC2307 is used by default.
search_timeout	No	Integer	LDAP search timeout interval, in seconds. If it is not specified, 3 seconds is used by default.
allow_local_user	No	String	Whether to allow local user authentication after access to the LDAP server failed.

Response Parameters

Status code: 200

Table 9-280 Response body parameters

Parameter	Type	Description
jobId	String	ID of an asynchronous LDAP task. You can query the task execution status by calling the API for querying details about a task.

Status code: 400

Table 9-281 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-282 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

Creating and binding an LDAP server

```
{
  "url" : "ldap://192.168.xx.xx:60000",
  "base_dn" : "dc=example,dc=com",
  "user_dn" : "cn=admin,dc=example,dc=com",
  "password" : "pwdxxxxx",
  "backup_url" : "ldap://192.168.xx.xx:60000",
  "schema" : "RFC2307",
  "search_timeout" : 10,
  "allow_local_user" : "Yes"
}
```

Example Responses

Status code: 200

Request accepted

```
{
  "jobId" : "72362dxxxxa04d419dbd5e6d9fe5xxxx"
}
```

Status code: 400

Client error

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "Invalid rule id"
}
```

Status code: 500

Internal error

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Request accepted
400	Client error
500	Internal error

Error Codes

See [Error Codes](#).

9.8.7 Querying the LDAP Configuration

Function

This API is used to query the LDAP configuration. Lightweight Directory Access Protocol (LDAP) is a standard protocol used for accessing and controlling directory servers. An LDAP server can centrally manage the relationship between users and groups. After an LDAP server is bound, when a user accesses a file in your file system, SFS Turbo accesses your LDAP server for user authentication and obtains the relationship between users and groups. In this way, standards Linux file UGO permissions are checked. To use this function, you need to first set up an LDAP server. SFS Turbo only supports LDAP v3 currently. Common directory servers that provide LDAP access include OpenLDAP (Linux) and Active Directory (Windows). The implementation varies depending on the directory server. When binding an LDAP server, you need to specify the corresponding schema. If the configured schema is incorrect, SFS Turbo cannot obtain the correct user and group information. As a result, users may fail to access files in the file system. Schemas that SFS Turbo supports include:

1. RFC2307 (Usually selected for OpenLDAP)
2. MS-AD-BIS (Usually selected for Active Directory. It supports RFC2307bis and nested groups.)

SFS Turbo also supports active and standby LDAP servers. If one LDAP server fails and cannot be accessed, SFS Turbo automatically switches to the standby LDAP server. In addition, if you set **allow_local_user** to **Yes** (default value is **No**), when both LDAP servers fail, SFS Turbo will use your local user and group information instead of the information configured on the LDAP server for identity authentication and UGO permissions check to minimize the fault impact.

URI

GET /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/ldap

Table 9-283 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-284 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-285 Response body parameters

Parameter	Type	Description
url	String	URL of the LDAP server
base_dn	String	Base DN
user_dn	String	User DN
filter_conditio n	String	Filter criteria. This is a reserved field and is not supported currently.
backup_url	String	URL of the standby LDAP server.
schema	String	LDAP schema. If it is not specified, RFC2307 is used by default.
search_timeo ut	Integer	LDAP search timeout interval, in seconds. If it is not specified, 3 seconds is used by default.
allow_local_us er	String	Whether to allow local user authentication after access to the LDAP server failed.

Status code: 500

Table 9-286 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

None

Example Responses

Status code: 200

Successful query

```
{
  "url" : "ldap://192.168.xx.xx:60000",
  "base_dn" : "dc=example,dc=com",
  "user_dn" : "cn=admin,dc=example,dc=com",
  "backup_url" : "ldap://192.168.xx.xx:60000",
  "schema" : "RFC2307",
  "search_timeout" : 10,
```

```
"allow_local_user" : "Yes"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Successful query
500	Error response

Error Codes

See [Error Codes](#).

9.8.8 Modifying the LDAP Configuration

Function

This API is used to modify the LDAP configuration. Lightweight Directory Access Protocol (LDAP) is a standard protocol used for accessing and controlling directory servers. An LDAP server can centrally manage the relationship between users and groups. After an LDAP server is bound, when a user accesses a file in your file system, SFS Turbo accesses your LDAP server for user authentication and obtains the relationship between users and groups. In this way, standards Linux file UGO permissions are checked. To use this function, you need to first set up an LDAP server. SFS Turbo only supports LDAP v3 currently. Common directory servers that provide LDAP access include OpenLDAP (Linux) and Active Directory (Windows). The implementation varies depending on the directory server. When binding an LDAP server, you need to specify the corresponding schema. If the configured schema is incorrect, SFS Turbo cannot obtain the correct user and group information. As a result, users may fail to access files in the file system. Schemas that SFS Turbo supports include:

1. RFC2307 (Usually selected for OpenLDAP)
2. MS-AD-BIS (Usually selected for Active Directory. It supports RFC2307bis and nested groups.)

SFS Turbo also supports active and standby LDAP servers. If one LDAP server fails and cannot be accessed, SFS Turbo automatically switches to the standby LDAP server. In addition, if you set **allow_local_user** to **Yes** (default value is **No**), when both LDAP servers fail, SFS Turbo will use your local user and group information instead of the information configured on the LDAP server for identity authentication and UGO permissions check to minimize the fault impact.

URI

PUT /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/ldap

Table 9-287 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-288 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Table 9-289 Request body parameters

Parameter	Mandatory	Type	Description
url	No	String	URL of the LDAP server. The format is <i>ldap://{ip_address}:{port_number}</i> or <i>ldaps://{ip_address}:{port_number}</i> , for example, ldap://192.168.xx.xx:60000 .
base_dn	No	String	Base DN
user_dn	No	String	User DN
password	No	String	LDAP authentication password
vpc_id	No	String	ID of the VPC
filter_condition	No	String	Filter criteria. This is a reserved field and is not supported currently.
backup_url	No	String	URL of the standby LDAP server. The format is <i>ldap://{ip_address}:{port_number}</i> or <i>ldaps://{ip_address}:{port_number}</i> , for example, ldap://192.168.xx.xx:60000 .

Parameter	Mandatory	Type	Description
schema	No	String	LDAP schema. If it is not specified, RFC2307 is used by default.
search_timeout	No	Integer	LDAP search timeout interval, in seconds. If it is not specified, 3 seconds is used by default.
allow_local_user	No	String	Whether to allow local user authentication after access to the LDAP server failed.

Response Parameters

Status code: 200

Table 9-290 Response body parameters

Parameter	Type	Description
jobId	String	ID of an asynchronous LDAP task. You can query the task execution status by calling the API for querying details about a task.

Status code: 400

Table 9-291 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-292 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

```
{
  "url" : "ldap://192.168.xx.xx:60000",
  "base_dn" : "dc=example,dc=com",
  "user_dn" : "cn=admin,dc=example,dc=com",
  "password" : "pwdxxxxx",
  "vpc_id" : "26f6b565-xxx-XXX-xxx-03f0bd975433",
  "backup_url" : "ldap://192.168.xx.xx:60000",
  "schema" : "RFC2307",
  "search_timeout" : 10,
  "allow_local_user" : "Yes"
}
```

Example Responses

Status code: 200

Updating

```
{
  "jobId" : "72362dxxxxa04d419dbd5e6d9fe5xxxx"
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "Invalid rule id"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Updating
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.8.9 Deleting the LDAP Configuration

Function

This API is used to delete the LDAP configuration. Lightweight Directory Access Protocol (LDAP) is a standard protocol used for accessing and controlling directory servers. An LDAP server can centrally manage the relationship between users and groups. After an LDAP server is bound, when a user accesses a file in your file system, SFS Turbo accesses your LDAP server for user authentication and obtains the relationship between users and groups. In this way, standards Linux file UGO permissions are checked. To use this function, you need to first set up an LDAP server. SFS Turbo only supports LDAP v3 currently. Common directory servers that provide LDAP access include OpenLDAP (Linux) and Active Directory (Windows). The implementation varies depending on the directory server. When binding an LDAP server, you need to specify the corresponding schema. If the configured schema is incorrect, SFS Turbo cannot obtain the correct user and group information. As a result, users may fail to access files in the file system. Schemas that SFS Turbo supports include:

1. RFC2307 (Usually selected for OpenLDAP)
2. MS-AD-BIS (Usually selected for Active Directory. It supports RFC2307bis and nested groups.)

SFS Turbo also supports active and standby LDAP servers. If one LDAP server fails and cannot be accessed, SFS Turbo automatically switches to the standby LDAP server. In addition, if you set **allow_local_user** to **Yes** (default value is **No**), when both LDAP servers fail, SFS Turbo will use your local user and group information instead of the information configured on the LDAP server for identity authentication and UGO permissions check to minimize the fault impact.

URI

DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/fs/ldap

Table 9-293 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
share_id	Yes	String	File system ID

Request Parameters

Table 9-294 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-295 Response body parameters

Parameter	Type	Description
jobId	String	ID of an asynchronous LDAP task. You can query the task execution status by calling the API for querying details about a task.

Status code: 400

Table 9-296 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-297 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

None

Example Responses

Status code: 200

Deleting

```
{
  "job_id" : "72362dxxxxa04d419dbd5e6d9fe5xxxx"
}
```

Status code: 400

Error response

```
{
  "errCode" : "SFS.TURBO.0001",
```

```
"errMsg" : "Invalid rule id"
}
```

Status code: 500

Error response

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Deleting
400	Error response
500	Error response

Error Codes

See [Error Codes](#).

9.9 Task Management

9.9.1 Querying Details About a Task

Function

This API is used to query the execution status of an SFS Turbo asynchronous task. For example, you can query the task execution status using the **jobId** returned after you call the API for creating and binding the LDAP configuration.

URI

GET /v1/{project_id}/sfs-turbo/jobs/{job_id}

Table 9-298 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
job_id	Yes	String	job ID

Request Parameters

Table 9-299 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Account token
Content-Type	Yes	String	MIME type

Response Parameters

Status code: 200

Table 9-300 Response header parameters

Parameter	Type	Description
X-request-id	String	Request ID

Table 9-301 Response body parameters

Parameter	Type	Description
status	String	Task status, which can be success , running , failed , or waiting
job_id	String	Task ID
job_type	String	Task type
begin_time	String	Task start time in UTC format, for example, '2016-01-02 15:04:05'
end_time	String	Task end time in UTC format, for example, '2016-01-02 15:04:05'
error_code	String	Error code returned if the task execution fails
fail_reason	String	Cause of the task execution failure
sub_jobs	Array of GetSubJobDe tail objects	List of subtasks

Table 9-302 GetSubJobDetail

Parameter	Type	Description
status	String	Subtask status, which can be success , running , failed , or waiting
job_id	String	Task ID
job_type	String	Subtask type
begin_time	String	Task start time in UTC format, for example, '2016-01-02 15:04:05'
end_time	String	Task end time in UTC format, for example, '2016-01-02 15:04:05'
error_code	String	Error code returned if the task execution fails
fail_reason	String	Cause of the task execution failure

Status code: 400

Table 9-303 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 404

Table 9-304 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Status code: 500

Table 9-305 Response body parameters

Parameter	Type	Description
errCode	String	Error code
errMsg	String	Error description

Example Requests

None

Example Responses

Status code: 200

Response body parameters

```
{
  "job_id" : "26f6b565-xxxx-XXXX-xxxx-03f0bd975433",
  "status" : "success",
  "job_type" : "bind_ldap",
  "begin_time" : "2023-07-26 09:33:58",
  "end_time" : "2023-07-26 09:33:58"
}
```

Status code: 400

Client error

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "parameter error"
}
```

Status code: 404

Resource not found

```
{
  "errCode" : "SFS.TURBO.0001",
  "errMsg" : "parameter error"
}
```

Status code: 500

Internal error

```
{
  "errCode" : "SFS.TURBO.0005",
  "errMsg" : "Internal server error"
}
```

Status Codes

Status Code	Description
200	Response body parameters
400	Client error
404	Resource not found
500	Internal error

Error Codes

See [Error Codes](#).

10 General Purpose File System APIs

10.1 File Systems

10.1.1 Creating a File System

Function

This API is used to create a file system.

URI

PUT /

Request Parameters

Table 10-1 Request header parameters

Parameter	Mandatory	Type	Description
Authorization	Yes	String	The signature information.
Date	Yes	String	The request time.
x-obs-az-redundancy	No	String	The AZ redundancy, single-AZ or multi-AZ.
x-obs-bucket-type	Yes	String	The header used to specify the file system creation. Enumerated value: <ul style="list-style-type: none">• SFS: creating a file system
Host	Yes	String	The host address.
x-obs-epid	No	String	The enterprise project ID.

Table 10-2 Request body parameter

Parameter	Mandatory	Type	Description
Location	No	String	The region.

Response Parameters

This response uses common headers. For details, see [Table 4-8](#).

(Optional) Response Body

A response body contains information other than the response header. It is usually sent in a structured format (JSON or XML) defined by the response header parameter **Content-type**.

Example Request

Creating a file system in example (with the host address and enterprise project ID 0):

```
PUT / HTTP/1.1
Host: example-sfs-01.sfs3.example.region.com:443
Date: Wed, 07 Jun 2023 02:38:09 GMT
x-obs-bucket-type: SFS
Authorization: OBS FNEX1B77SXDIB3TFMSZZ:0Xsnu4hJVOI7VWH0wIQczVN+rbg=
Content-Length: 85
x-obs-epid: 0

<CreateBucketConfiguration>
  <Location>example</Location>
</CreateBucketConfiguration>
```

Example Response

```
HTTP/1.1 200 OK
Server: OBS
X-Obs-Request-Id: 0000018893B8058EC0470388BE6EDE88
Location: /example-sfs-01
X-Obs-Id-2: 32AAAQAAEAABSAAgAAEAABAAAQAAEAABCTRa4voOUvr50ncznQT/hligMxL4so2z
Date: Wed, 07 Jun 2023 02:38:11 GMT
Content-Length: 0
```

Status Codes

Status Code	Description
200	The file system is created.

Error Codes

See [General Purpose File System Error Codes](#).

10.1.2 Deleting a File System

Function

This API is used to delete a file system.

URI

DELETE /

Request Parameters

Table 10-3 Request header parameters

Parameter	Mandatory	Type	Description
Authorization	Yes	String	The signature header field.
Date	Yes	String	The request time.
Host	Yes	String	The host address.

Response Parameters

This response uses common headers. For details, see [Table 4-8](#).

Example Request

```
DELETE / HTTP/1.1
User-Agent: curl/7.29.0
Accept: */*
Date: WED, 01 Jul 2015 02:31:25 GMT
Authorization: OBS H4IPJX0TQTHTHEBQQCEC:jZiAT8Vx4azWEvPRMWi0X5BpJMA=
```

Example Response

```
HTTP/1.1 204 No Content
Server: OBS
X-Obs-Request-Id: 0000018893B8081DC047305E783867DD
X-Obs-Id-2: 32AAAQAAEAABSkAgAAEAABAAAQAAEAABCT5UWgsaro3EEEnOsNEzf8w8dnydR+Eak
Date: WED, 01 Jul 2015 02:31:25 GMT
```

Status Codes

Status Code	Description
204	The file system is deleted.

Error Codes

See [General Purpose File System Error Codes](#).

10.1.3 Listing File Systems

Function

This API is used to list file systems.

URI

GET /

Request Parameters

Table 10-4 Request header parameters

Parameter	Mandatory	Type	Description
x-obs-bucket-type	Yes	String	This header field is used to specify the content to be obtained. Enumerated value: <ul style="list-style-type: none"> • SFS: listing all file systems
Authorization	Yes	String	The signature information.
Date	Yes	String	The request time.
Host	Yes	String	The host address.

Response Parameters

Status code: 200

Table 10-5 Response body parameters

Parameter	Type	Description
Owner	Owner object	File system owner information, including the tenant ID.
Buckets	Buckets object	The list of file systems owned by the user.

Table 10-6 Owner

Parameter	Type	Description
ID	String	The domain ID (account ID) of a user.

Table 10-7 Buckets

Parameter	Type	Description
Bucket	Bucket object	Detailed file system information.

Table 10-8 Bucket

Parameter	Type	Description
Name	String	The name of a file system.
CreationDate	String	The time when a file system was created.
Location	String	The location of a file system.

Example Request

```
GET / HTTP/1.1 Date: date
x-obs-bucket-type: SFS
Authorization: authorization
```

Example Response

```
HTTP/1.1 200 OK
Server: OBS
X-Obs-Request-Id: 0000018893B8126DC048B06DD3816BD4
X-Obs-Id-2: 32AAAQAAEAABAAAQAAEAABAAAQAAEAABCTMZh3Thi7lcDxuGWu9Qtp9PJbYXa7lb
Date: Wed, 07 Jun 2023 02:38:14 GMT
Content-Type: application/xml
Content-Length: 377

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ListAllMyBucketsResult xmlns="http://obs.myhwclouds.com/doc/2015-06-30/">
  <Owner>
    <ID>783fc6652cf246c096ea836694f71855</ID>
  </Owner>
  <Buckets>
    <Bucket>
      <Name>examplebucket01</Name>
      <CreationDate>2018-06-21T09:15:01.032Z</CreationDate>

      <Location>example-region-1</Location>
      <BucketType>SFS</BucketType>
    </Bucket>
    <Bucket>
      <Name>examplebucket02</Name>
      <CreationDate>2018-06-22T03:56:33.700Z</CreationDate>

      <Location>example-region-2</Location>
      <BucketType>SFS</BucketType>
    </Bucket>
  </Buckets>
</ListAllMyBucketsResult>
```

Status Codes

Status Code	Description
200	The file systems are obtained.

Error Codes

See [General Purpose File System Error Codes](#).

10.2 File System Access Rules

10.2.1 Configuring a File System ACL

Function

This API is used to configure a file system ACL.

NOTE

After the ACL is configured, the configuration takes about 30 second to take effect.

URI

PUT /

Table 10-9 Query parameter

Parameter	Mandatory	Type	Description
sfsacl	Yes	String	/

Request Parameters

Table 10-10 Request header parameters

Parameter	Mandatory	Type	Description
Date	Yes	String	The request time.
Authorization	Yes	String	The signature information.
Host	Yes	String	The host address.

Table 10-11 Request body parameter

Parameter	Mandatory	Type	Description
Statement	No	Array of Statement objects	Unique identification

Table 10-12 Statement

Parameter	Mandatory	Type	Description
Sid	No	String	The statement ID.
Action	Yes	String	The allowed statement action. Enumerated values: <ul style="list-style-type: none"> • FullControl: read/write • Read: read-only
Effect	Yes	String	The effect specifying that the statement permission is Allow . Enumerated value: <ul style="list-style-type: none"> • Allow
Condition	Yes	Condition object	The conditions for a statement to take effect.

Table 10-13 Condition

Parameter	Mandatory	Type	Description
SourceVpc	Yes	String	A specified VPC ID.
VpcSourceIps	No	Array of strings	A specified IP address or IP address range. This parameter is currently not supported.

Response Parameters

This response uses common headers. For details, see [Table 4-8](#).

Example Request

Configuring a file system ACL (granting the read/write permissions for IP addresses **127.0.0.1/24** and **192.168.1.85/24** in VPC **241dbf6b-dc5d-41b2-9108-ca5e56b48386**):

```
PUT /?sfsacl HTTP/1.1
Host: examplefilesystem.sfs3.example.region.com
```

```
Date: WED, 01 Jul 2015 02:32:25 GMT
Authorization: OBS H4IPJX0TQTHTHEBQQCEC:jZiAT8Vx4azWEvPRMWi0X5BpJMA=
{
  "Statement": [{
    "Sid": "Stmt1375240018061",
    "Action": "FullControl",
    "Effect": "Allow",
    "Condition": {
      "SourceVpc": "241dbf6b-dc5d-41b2-9108-ca5e56b48386",
      "VpcSourceIp": ["127.0.0.1/24", "192.168.1.85/24"]
    }
  }]
}
```

Example Response

```
HTTP/1.1 204 OK
Server: OBS
X-Obs-Request-Id: 0000018893B8073AC04721AA7EE3408B
X-Obs-Id-2: 32AAAQAAEAABSAAgAAEAABAAAQAAEAABCS5QDe0QLbFNz6FXoKuXHzD2wS0eJQaj
Date: Wed, 07 Jun 2023 02:38:11 GMT
```

Status Codes

Status Code	Description
204	The file system ACL is configured.

Error Codes

See [General Purpose File System Error Codes](#).

10.2.2 Obtaining File System ACL Information

Function

This API is used to obtain the ACL information of a file system.

URI

GET /

Table 10-14 Query parameter

Parameter	Mandatory	Type	Description
sfsacl	Yes	String	/

Request Parameters

Table 10-15 Request header parameters

Parameter	Mandatory	Type	Description
Date	Yes	String	The request time.
Authorization	Yes	String	The signature information.
Host	Yes	String	The host address.

Response Parameters

Status code: 200

Table 10-16 Response body parameter

Parameter	Type	Description
Statement	Array of Statement objects	Unique identification

Table 10-17 Statement

Parameter	Type	Description
Sid	String	The statement ID.
Action	String	The allowed statement action. Enumerated values: <ul style="list-style-type: none"> • FullControl: read/write • Read: read-only
Effect	String	The effect specifying that the statement permission is Allow . Enumerated value: <ul style="list-style-type: none"> • Allow
Condition	Condition object	The conditions for a statement to take effect.

Table 10-18 Condition

Parameter	Type	Description
SourceVpc	String	A specified VPC ID.

Parameter	Type	Description
VpcSourceIp	Array of strings	A specified IP address or IP address range. This parameter is currently not supported.

Example Request

```
GET /?sfsacl HTTP/1.1
Host: example-sfs-01.sfs3.example.region.com:443
Date: Wed, 07 Jun 2023 03:31:46 GMT
Authorization: OBS FNEX1B77SXDIB3TFMSZZ:eUqPlHnPDWGDTlgyLmsALA86wys=
```

Example Response

```
HTTP/1.1 200 OK
Server: OBS
Content-Type: application/json
Content-Length: 131
Date: Wed, 07 Jun 2023 03:31:59 GMT
X-Obs-Request-Id: 0000018893E94B65C046B527778F8F14
X-Obs-Id-2: 32AAAQAAEAABAAAQAAEAABAAAQAAEAABCSc2lEdSHcA04319WknB1DD5BdBKuGr1
{
  "Statement": [
    {
      "Condition": {
        "SourceVpc": "f85adabc-a387-4d1d-94cf-65ef9034f752"
      },
      "Action": "FullControl",
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
```

Status Codes

Status Code	Description
200	The file system ACL is obtained.

Error Codes

See [General Purpose File System Error Codes](#).

10.2.3 Deleting a File System ACL

Function

This API is used to delete a file system ACL.

URI

DELETE /

Table 10-19 Query parameter

Parameter	Mandatory	Type	Description
sfsacl	Yes	String	/

Request Parameters

Table 10-20 Request header parameters

Parameter	Mandatory	Type	Description
Date	Yes	String	The request time.
Authorization	Yes	String	The signature information.
Host	Yes	String	The host address.

Response Parameters

This response uses common headers. For details, see [Table 4-8](#).

Example Request

```
DELETE /?sfsacl HTTP/1.1
Host: examplefilesystem.sfs3.example.region.com
Date: WED, 01 Jul 2015 02:36:06 GMT
Authorization: OBS H4IPJX0TQTHTHEBQQCEC;jZiAT8Vx4azWEvPRMWi0X5BpJMA=
```

Example Response

```
HTTP/1.1 204 No Content
Server: OBS
X-Obs-Id-2: 32AAAQAAEAABSAAgAAEAABAAAQAAEAABCSj4dxiqb1Lw50CTjVQeV3ebh3QQ6PAj
X-Obs-Request-Id: 0000018893B807D5C0472A6161D87032
Date: WED, 01 Jul 2015 02:36:06 GMT
```

Status Codes

Status Code	Description
204	The file system ACL is deleted.

Error Codes

See [General Purpose File System Error Codes](#).

10.3 Tags

10.3.1 Batch Adding Tags to a Resource

Function

This API is used to batch add tags for a general purpose file system. You can add up to 20 tags to a resource.

URI

- POST /v3/sfs/tms/{project_id}/file-systems/{resource_id}/tags/create
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .
resource_id	Yes	String	The resource ID, which is the name of a general purpose file system.

Request Parameters

Table 10-21 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json
X-Auth-Token	No	String	The user token.

Table 10-22 Request body parameters

Parameter	Mandatory	Type	Description
tags	No	List< resource_tag >	The tag list. For details, see Table 10-23 . This parameter is mandatory for common tenants. Use either tags or sys_tags if you have the op_service permissions.

Parameter	Mandatory	Type	Description
sys_tags	No	List< resource_tag >	<p>The system tag list. This parameter is available only to the op_service permissions.</p> <p>Use either tags or sys_tags if you have the op_service permissions.</p> <p>Only one resource_tag structure is used in TMS calls currently.</p> <p>The key is fixed at _sys_enterprise_project_id.</p> <p>The value can be UUID or 0. 0 indicates the default enterprise project.</p> <p>System tags can only be added.</p> <p>For details, see Table 10-23.</p>

Table 10-23 resource_tag

Parameter	Mandatory	Type	Description
key	Yes	String	<p>The tag key. A tag key can contain a maximum of 128 characters. It can contain letters, digits, and spaces representable in UTF-8 and special characters (<code>_.:=+@</code>). It cannot start or end with a space and cannot be left empty. Tag keys starting with _sys_ are system tags, and you cannot start a tag key with _sys_.</p>
value	No	String	<p>The tag value. A tag value can contain a maximum of 255 characters. It can contain letters, digits, and spaces representable in UTF-8 and special characters (<code>_.:=+@</code>) and can be left empty. It cannot start or end with a space.</p>

Response Parameters

None

Example Request

Batch adding tags to a general purpose file system whose name is **bucketName** with the project ID **c80a2157ba1d46c0825265947342077c**:

```
POST https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/bucketName/tags/create
```

Request body example:

```
{
  "tags": [
    {
      "key": "key1",
      "value": "value1"
    },
    {
      "key": "key2",
      "value": "value2"
    }
  ]
}
```

Example Response

None

Status Codes

- Normal

Status Code	Description
204	Resource tags added.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

10.3.2 Batch Deleting Tags from a Resource

Function

This API is used to batch delete tags from a specified resource. System tags cannot be deleted. If any tag to be deleted is not found, the deletion is considered successful.

URI

- POST /v3/sfs/tms/{project_id}/file-systems/{resource_id}/tags/delete
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .
resource_id	Yes	String	The resource ID, which is the name of a general purpose file system.

Request Parameters

Table 10-24 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json
X-Auth-Token	No	String	The user token.

Table 10-25 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	List< resource_tag >	The tag list. For details, see Table 10-23 .

Response Parameters

None

Example Request

Batch deleting tags from a general purpose file system whose name is **bucketName** with the project ID **c80a2157ba1d46c0825265947342077c**:

POST https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/bucketName/tags/delete

Request body example:

```
{
  "tags":[
    {
      "key":"key1"
    },
    {
      "key":"key2",
      "value":"value2"
    }
  ]
}
```

Example Response

None

Status Codes

- Normal

Status Code	Description
204	Resource tags deleted.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

10.3.3 Querying Tags of a Resource

Function

This API is used to query tags of a specified resource.

URI

- GET /v3/sfs/tms/{project_id}/file-systems/{resource_id}/tags

- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .
resource_id	Yes	String	The resource ID, which is the name of a general purpose file system.

Request Parameters

Table 10-26 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json
X-Auth-Token	No	String	The user token.

Response Parameters

Table 10-27 Response body parameters

Parameter	Mandatory	Type	Description
tags	No	List< resource_tag >	The tag list. For details, see Table 10-23 .
sys_tags	No	List< resource_tag >	The system tag list. This parameter is only available to users with the op_service permission. This field contains only one resource_tag structure key currently. The key is fixed at _sys_enterprise_project_id . The value is the ID of an enterprise project. Value 0 indicates the default enterprise project. For details, see Table 10-23 .

Example Request

Querying tags of a general purpose file system whose name is **bucketName** with the project ID **c80a2157ba1d46c0825265947342077c**:

GET `https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/bucketName/tags`

Example Response

```
{
  "tags":[
    {
      "key":"key1",
      "value":"value1"
    },
    {
      "key":"key2",
      "value":"value2"
    }
  ]
}
```

Status Codes

- Normal

Status Code	Description
200	Resource tags queried.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

10.3.4 Listing Resources

Function

This API is used to list resources by tag. Resources are sorted by the time when they are created, in descending order.

URI

- POST /v3/sfs/tms/{project_id}/file-systems/resource-instances/filter
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .
limit	No	Int	The number of records to be queried. The value ranges from 1 to 1000 , and the default value is 1000 .
offset	No	Int	The index location. The query starts from the next data record indexed by this parameter. The default value is 0 , indicating that the query starts from the first data record. The value cannot be a negative number.

Request Parameters

Table 10-28 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json
X-Auth-Token	No	String	The user token.

Table 10-29 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	boolean	Excludes any of the specified tags. If this parameter is set to true , all resources without tags are queried. In this case, the tags field is ignored. If this parameter is set to false or not specified, it does not take effect, which means that all resources are returned or resources are filtered by tags or matches .

Parameter	Mandatory	Type	Description
tags	No	List<tag>	<p>Includes specified tags. A maximum of 20 tags can be specified. Tag keys must be unique. Each tag key can have a maximum of 20 tag values. A tag value can be empty but the structure cannot be missing. Tag values of the same key must be unique.</p> <p>The response returns resources containing all tags in this list. Keys in this list are in an AND relationship while values in each key-value structure is in an OR relationship. If this parameter is not specified, all resources will be returned.</p> <p>For details, see Table 10-30.</p>
sys_tags	No	List<tag>	<p>Contains system tags. This parameter is only available to users with the op_service permission.</p> <p>This parameter cannot be used with filtering criteria without_any_tag and tags at the same time.</p> <p>This field contains only one tag structure currently.</p> <p>The key is fixed at _sys_enterprise_project_id.</p> <p>The value is the ID of an enterprise project.</p> <p>The key contains only one value. Value 0 indicates the default enterprise project.</p> <p>For details, see Table 10-30.</p>
matches	No	List<match>	<p>The fields to be matched. The key in match is a dictionary value fixed at resource_name, meaning that the prefix search is performed based on the value of the key. It will be extended later.</p> <p>For details, see Table 10-31.</p>

Table 10-30 tag

Parameter	Mandatory	Type	Description
key	Yes	String	The tag key. A tag key can contain a maximum of 128 characters. It can contain letters, digits, and spaces representable in UTF-8 and special characters (<code>_:=+~@</code>). It cannot start or end with a space and cannot be left empty. Tag keys starting with <code>_sys_</code> are system tags, and you cannot start a tag key with <code>_sys_</code> .
values	Yes	List<String>	The tag value list. A value can be an empty array but cannot be left blank. If the list specified by values is empty arrays, any tag value can be queried. The values are in the OR relationship.

Table 10-31 match

Parameter	Mandatory	Type	Description
key	Yes	String	The key. The key is fixed at resource_name currently. Other key values will be available later.
value	Yes	String	The value. Each tag value can contain a maximum of 255 Unicode characters. The character set is not verified.

Response Parameters

Table 10-32 Response body parameters

Parameter	Mandatory	Type	Description
resources	Yes	List< resource >	The resource list. For details, see Table 10-33 .
total_count	Yes	Integer	The total number of records.

Table 10-33 resource

Parameter	Mandatory	Type	Description
resource_id	Yes	String	The resource ID.
resource_detail	Yes	Object	The resource details. This value is left empty by default. The value is a resource object used for extension.
tags	Yes	List<resource_tag>	The tag list. If there is no tag, an empty array is used by default. For details, see Table 10-23 .
sys_tags	No	List<resource_tag>	This parameter is only available to users with the op_service permission. This field contains only one resource_tag structure key currently. The key is fixed at _sys_enterprise_project_id . The value is the ID of the enterprise project. Value 0 indicates the default enterprise project. For details, see Table 10-23 .
resource_name	Yes	String	The resource name. This parameter is an empty string if there is no resource name.

Example Request

Listing resources (with the project ID set to ****c80a2157ba1d46c0825265947342077c****, offset to ****0****, and limit to ****10****):

POST <https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/resource-instances/filter?limit=10&offset=0>

Request body example:

```
{
  "tags":[
    {
      "key":"key1",
      "values":[
        "value1",
        "value2"
      ]
    },
    {
      "key":"key2",
      "values":[]
    }
  ]
}
```



```

        "value1",
        "value2"
    ]
  },
  "matches":[
    {
      "key":"resource_name",
      "value":"resource1"
    }
  ],
  "without_any_tag":"false"
}

```

Example Response

```

{
  "resources":[
    {
      "resource_detail":"","
      "resource_id":"resouce1",
      "resource_name":"resouce1",
      "tags":[
        {
          "key":"key1",
          "value":"value1"
        }
      ],
      "sys_tags":[]
    }
  ],
  "total_count":1
}

```

Status Codes

- Normal

Status Code	Description
200	Resources listed.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

10.3.5 Querying the Number of Resources

Function

This API is used to query the number of resources.

URI

- POST /v3/sfs/tms/{project_id}/file-systems/resource-instances/count
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .

Request Parameters

Table 10-34 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json
X-Auth-Token	No	String	The user token.

Table 10-35 Request body parameters

Parameter	Mandatory	Type	Description
without_any_tag	No	boolean	Excludes resources with any of the specified tags. If this parameter is set to true , all resources without tags are queried. In this case, the tags field is ignored. If this parameter is set to false or not specified, it does not take effect, which means that all resources are returned or resources are filtered by tags or matches .

Parameter	Mandatory	Type	Description
tags	No	List<tag>	The resources to be queried contain tags listed in tags . Each resource to be queried can contain a maximum of 20 tag keys, and each key can have a maximum of 20 values. The tag value of a tag key can be an empty array but the structure cannot be missing. Each tag key must be unique, and the tag values of a key must also be unique. The response returns resources containing all tags in this list. Keys in this list are in an AND relationship while values in each key-value structure is in an OR relationship. If this parameter is not specified, all resources will be returned. For details, see Table 10-30 .
sys_tags	No	List<tag>	Contains system tags. This parameter is only available to users with the op_service permission. This parameter cannot be used with filtering criteria without_any_tag and tags at the same time. This field contains only one tag structure currently. The key is fixed at _sys_enterprise_project_id . The value is the ID of an enterprise project. The key contains only one value. Value 0 indicates the default enterprise project. For details, see Table 10-30 .
matches	No	List<match>	The fields to be matched. The key in match is a dictionary value fixed at resource_name , meaning that the prefix search is performed based on the value of the key. It will be extended later. For details, see Table 10-31 .

Response Parameters

Table 10-36 Response body parameter

Parameter	Mandatory	Type	Description
total_count	Yes	Integer	The total number of records.

Example Request

Querying the number of resources using project ID
c80a2157ba1d46c0825265947342077c:

POST https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/resource-instances/count

Request body example:

```
{
  "tags":[
    {
      "key":"key1",
      "values":[
        "value1",
        "value2"
      ]
    },
    {
      "key":"key2",
      "values":[
        "value1",
        "value2"
      ]
    }
  ],
  "matches":[
    {
      "key":"resource_name",
      "value":"resource1"
    }
  ],
  "without_any_tag":"true"
}
```

Example Response

```
{
  "total_count":1
}
```

Status Codes

- Normal

Status Code	Description
200	Number of resources queried.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

10.3.6 Querying Tags by Project

Function

This API is used to query tags of all resources owned by a tenant in a specified project.

URI

- GET /v3/sfs/tms/{project_id}/file-systems/tags
- Parameter description

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details, see Obtaining a Project ID .

Request Parameters

Table 10-37 Request header parameters

Parameter	Mandatory	Type	Description
Content-type	Yes	String	The MIME type of the request body. Example: application/json

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	The user token.

Response Parameters

Table 10-38 Response body parameter

Parameter	Mandatory	Type	Description
tags	Yes	List<tag>	The tag list. For details, see Table 10-30 .

Example Request

Querying tags using project ID ****c80a2157ba1d46c0825265947342077c****:

```
GET https://{endpoint}/v3/sfs/tms/c80a2157ba1d46c0825265947342077c/file-systems/tags
```

Example Response

```
{
  "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    },
    {
      "key": "key2",
      "values": [
        "value1",
        "value2"
      ]
    }
  ]
}
```

Status Codes

- Normal

Status Code	Description
200	Resource tags queried by project.

- Abnormal

Status Code	Description
400	Invalid tag parameter.
401	Certification failed.
403	Authentication failed.
404	Resource not found.
500	System error.

11 Permissions Policies and Supported Actions

11.1 Introduction

This section describes fine-grained permissions management for your SFS resources. If your Huawei Cloud account does not need individual IAM users, then you may skip over this section.

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

You can grant users permissions by using roles and policies. Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

- Policy-based authorization is useful if you want to allow or deny the access to an API.

Each account has all the permissions required to call all APIs, but IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user wants to query ECSs using an API, the user must have been granted permissions that allow the **ecs:servers:list** action.

Supported Actions

SFS provides system-defined policies that can be directly used in IAM. You can also create custom policies and use them to supplement system-defined policies, implementing more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Statements in a policy that allow or deny certain operations.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.
- Actions: Specific operations that are allowed or denied.
- Related actions: Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the related actions.
- IAM projects/Enterprise projects: Authorization scope of a custom policy. A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions for both IAM and enterprise projects can be used and applied for both IAM and Enterprise Management. Policies that only contain actions for IAM projects can be used and applied to IAM only. For details about the differences between IAM and enterprise projects, see [Differences Between IAM and Enterprise Management](#).

•  **NOTE**

The check mark (√) and cross symbol (x) indicate that an action takes effect or does not take effect for the corresponding type of projects.

SFS supports the following actions that can be defined in custom policies:

- **Shared File Systems**, including actions supported by all SFS file system APIs, such as the APIs for creating file systems, querying file system lists, querying details about a single file system, modifying file systems, and deleting file systems.
- **Capacity Expansion or Reduction**, including actions supported by the SFS file system expanding and shrinking APIs, such as the APIs for expanding or shrinking a shared file system.
- **SFS Turbo Actions**, including actions supported by all SFS Turbo file system APIs, such as the APIs for creating file systems, querying file system lists, querying details about a single file system, and deleting file systems.

11.2 Supported Actions

11.2.1 SFS Actions

API Version Querying

Permission	API	Action	IAM Project	Enterprise Project
Querying the API Version (Native OpenStack API)	GET /	-	x	x

Permission	API	Action	IAM Project	Enterprise Project
Querying the API Version (Native OpenStack API)	GET /{api_version}/	-	×	×

Shared File Systems

Permission	API	Action	IAM Project	Enterprise Project
Creating a Shared File System (Native OpenStack API)	POST /v2/{project_id}/shares	sfs:shares:createShare	√	√
Querying All Shared File Systems (Native OpenStack API)	GET /v2/{project_id}/shares	sfs:shares:getShare	√	√
Querying Details About All Shared File Systems (Native OpenStack API)	GET /v2/{project_id}/shares/detail	sfs:shares:getAllSharesDetail	√	√
Querying Details About a Shared File System (Native OpenStack API)	GET /v2/{project_id}/shares/{share_id}	sfs:shares:getShareDetail	√	√

Permission	API	Action	IAM Project	Enterprise Project
Querying Mount Locations of a Shared File System (Native OpenStack API)	GET /v2/{project_id}/shares/{share_id}/export_locations	sfs:shares:getShareExportLocations	√	√
Modifying a Shared File System (Native OpenStack API)	PUT /v2/{project_id}/shares/{share_id}	sfs:shares:updateShare	√	√
Deleting a Shared File System (Native OpenStack API)	DELETE /v2/{project_id}/shares/{share_id}	sfs:shares:deleteShare	√	√

Share Access Rules

Permission	API	Action	IAM Project	Enterprise Project
Adding Share Access Rules (Native OpenStack API)	POST /v2/{project_id}/shares/{share_id}/action	sfs:shares:ShareAction	√	√
Deleting Share Access Rules (Native OpenStack API)	POST /v2/{project_id}/shares/{share_id}/action	sfs:shares:ShareAction	√	√

Permission	API	Action	IAM Project	Enterprise Project
Querying Share Access Rules (Native OpenStack API)	POST /v2/{project_id}/shares/{share_id}/action	sfs:shares:ShareAction	√	√

Quota Management

Permission	API	Action	IAM Project	Enterprise Project
Querying the Tenant Quota (Native OpenStack API)	GET /v2/{project_id}/os-quota-sets/{project_id}	sfs:quotas:getOSQuotaSets	√	×

Capacity Expansion or Reduction

Permission	API	Action	IAM Project	Enterprise Project
Expanding a Shared File System (Native OpenStack API)	POST /v2/{project_id}/shares/{share_id}/action	sfs:shares:ShareAction	√	√
Shrinking a Shared File System (Native OpenStack API)	POST /v2/{project_id}/shares/{share_id}/action	sfs:shares:ShareAction	√	√

Tags for a Shared File System

Permission	API	Action	IAM Project	Enterprise Project
Adding a Tag to a Shared File System	POST /v2/{project_id}/sfs/{share_id}/tags	sfs:tags:addShareTags	√	×
Deleting a Tag from a Shared File System	DELETE /v2/{project_id}/sfs/{share_id}/tags/{key}	sfs:tags:deleteShareTags	√	×
Querying Tag Information About a Shared File System	GET /v2/{project_id}/sfs/{share_id}/tags	sfs:tags:getShareTags	√	×
Querying Tags of a Tenant's All Shared File Systems	GET /v2/{project_id}/sfs/tags	sfs:tags:getAllTags	√	×
Batch Adding Tags to a Shared File System	POST /v2/{project_id}/sfs/{share_id}/tags/action	sfs:tags:batchShareTags	√	×
Batch Deleting Tags from a Shared File System	POST /v2/{project_id}/sfs/{share_id}/tags/action	sfs:tags:batchShareTags	√	×
Querying Details About a Shared File System Based on Tags	POST /v2/{project_id}/sfs/resource_instances/action	sfs:tags:getShareByTags	√	×

Permission	API	Action	IAM Project	Enterprise Project
Querying the Number of Shared File Systems Based on Tags	POST /v2/{project_id}/sfs/resource_instances/action	sfs:tags:getShareByTags	√	×

AZ

Permission	API	Action	IAM Project	Enterprise Project
Querying AZs (Native OpenStack API)	GET /v2/{project_id}/availability-zones?share_az={share_az}	sfs:availabilityZones:getAvailabilityZones	√	×

11.2.2 SFS Turbo Actions

File System

Permission	API	Action	Dependencies	IAM Project (Project)	Enterprise Project (Enterprise Project)
Creating a File System	POST /v1/{project_id}/sfs-turbo/shares	sfsturbo:shares:createShare	<ul style="list-style-type: none"> Creating an SFS Turbo file system requires VPC-related permissions, including the permissions for verifying VPCs, subnets, and security groups, creating virtual IP addresses and ports, and creating security group rules. <ul style="list-style-type: none"> "vpc:*:*" Creating an encrypted SFS Turbo file system requires the KMS Administrator permissions. Creating an SFS Turbo file system that uses DSS <ul style="list-style-type: none"> "dss:*:get", "dss:*:list", "dss:*:count" 	√	√

Permission	API	Action	Dependencies	IAM Project (Project)	Enterprise Project (Enterprise Project)
Querying Details About All File Systems	GET /v1/{project_id}/sfs-turbo/shares/detail	sfsturbo:shares:getAllShares	-	√	√
Querying Details About a File System	GET /v1/{project_id}/sfs-turbo/shares/{share_id}	sfsturbo:shares:getShare	-	√	√
Deleting a File System	DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}	sfsturbo:shares:deleteShare	<ul style="list-style-type: none"> Deleting an SFS Turbo file system requires VPC-related permissions, including the permissions for deleting virtual IP addresses and ports and deleting security group rules. <ul style="list-style-type: none"> "vpc:*:*" Deleting an SFS Turbo file system that uses DSS <ul style="list-style-type: none"> "dss:*:get", "dss:*:list", "dss:*:count" 	√	√
Adding a Storage Backend	POST /v1/{project_id}/sfs-turbo/shares/{share_id}/targets	sfsturbo:shares:createBackendTarget	You must have the OBS Administrator permissions.	√	√

Permission	API	Action	Dependencies	IAM Project (Project)	Enterprise Project (Enterprise Project)
Listing Storage Backends	GET /v1/{project_id}/sfs-turbo/shares/{share_id}/targets	sfsturbo:share s:listBackendTargets	-	√	√
Querying Details of a Storage Backend	GET /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}	sfsturbo:share s:showBackendTargetInfo	-	√	√
Removing a Storage Backend	DELETE /v1/{project_id}/sfs-turbo/shares/{share_id}/targets/{target_id}	sfsturbo:share s:deleteBackendTarget	-	√	√
Creating an Import or Export Task	POST /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task	sfsturbo:share s:createDataRepositoryTask	-	√	√
Querying Details About an Import or Export Task	GET /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/task/{task_id}	sfsturbo:share s:getDataRepositoryTask	-	√	√
Listing Import and Export Tasks	GET /v1/{project_id}/sfs-turbo/{share_id}/hpc-cache/tasks	sfsturbo:share s:getAllDataRepositoryTasks	-	√	√

File System Expansion

Permission	API	Action	IAM Project (Project)	Enterprise Project (Enterprise Project)
Expanding the Capacity of a File System	POST /v1/{project_id}/sfs-turbo/shares/{share_id}/action	sfsturbo:shares:shareAction	√	√

Console Reference

Permission	API	Action	Dependencies	IAM Project (Project)	Enterprise Project (Enterprise Project)
Changing a Security Group	Console reference	sfsturbo:shares:shareAction	Changing a security group <ul style="list-style-type: none"> vpc:securityGroups:* vpc:securityGroupRules:* 	√	√
Querying the SFS Turbo Quota	Console reference	sfsturbo:shares:getQuota	-	√	√
Obtaining the AZ Information	Console reference	sfsturbo:shares:getAZInfo	-	√	√
Obtaining SFS Turbo Specifications	Console reference	sfsturbo:shares:getFlavors	-	√	√
Checking the Name of a File System	Console reference	sfsturbo:shares:checkShareName	-	√	√

12 Common Parameters

12.1 SFS Turbo File System Statuses

- SFS Turbo file system status elements

Returned Value	Description
100	CREATING: The file system is being created.
200	ACTIVE: The file system is active. An SFS Turbo file system can be mounted in this status.
300	FAILED: The job failed.
303	CREATE_FAILED: The cluster failed to be created.
400	DELETED: The cluster has been deleted.
800	FROZEN: The cluster has been frozen.

12.2 SFS Turbo File System Substatuses

- SFS Turbo file system substatus elements

Returned Value	Description
121	Expanding the capacity online.
221	Online capacity expansion succeeded.
321	Failed to perform online capacity expansion.

13 Appendix

13.1 Status Codes

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

- Normal

Returned Value	Description
200 OK	Specifies the normal response for the GET and PUT operations.
201 Created	Specifies the normal response for the POST operation.
202 Accepted	The request has been accepted for processing.
204 No Content	Specifies the normal response for the DELETE operation.

- Abnormal

Returned Value	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and the password to access the requested page.
403 Forbidden	Access to the requested page is forbidden.
404 Not Found	The requested page was not found.
405 Method Not Allowed	You are not allowed to use the method specified in the request.

Returned Value	Description
406 Not Acceptable	The response generated by the server could not be accepted by the client.
407 Proxy Authentication Required	You must use the proxy server for authentication. Then the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	The request is not completed because of a service error.
501 Not Implemented	The request is not completed because the server does not support the requested function.
502 Bad Gateway	The request is not completed because the server receives an invalid response from an upstream server.
503 Service Unavailable	The request is not completed because the service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

13.2 SFS Turbo Error Codes

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

Status Code	Error Code	Error Message	Description	Solution
400/404	SFS.TURBO.0001	Parameter error	Invalid parameters.	Use valid parameters and try again.
400/404	SFS.TURBO.0002	Cluster not found	The requested object is not found or you do not have permissions to access it.	Use valid parameters and try again.
400	SFS.TURBO.0003	Invalid name	Invalid name.	Use valid parameters and try again.

Status Code	Error Code	Error Message	Description	Solution
400	SFS.TURBO.0004	Invalid vpc	Invalid VPC.	Use valid parameters and try again.
400/500	SFS.TURBO.0005	Internal server error	Internal error.	Contact technical support.
400	SFS.TURBO.0006	Invalid subnet	Invalid subnet.	Use valid parameters and try again.
400	SFS.TURBO.0007	Invalid share type	Invalid file system type.	Use valid parameters and try again.
400	SFS.TURBO.0008	Invalid size	Unsupported file system size.	Use valid parameters and try again.
409	SFS.TURBO.0009	Name has existed	File system name already exists.	Use valid parameters and try again.
400	SFS.TURBO.0010	Quota exceeds	Insufficient quota.	Submit a service order to increase quota.
400/403	SFS.TURBO.0011	Cluster is doing something	Another operation is being performed on the file system.	Wait until that operation is complete and try again.
400	SFS.TURBO.0012	Operation is not allowed	A yearly/monthly file system cannot be resized or deleted via API.	Manage yearly/monthly file systems on the console.
400	SFS.TURBO.0015	do not have the operation permission	Insufficient permissions.	Apply for the required permissions.

Status Code	Error Code	Error Message	Description	Solution
400	SFS.TURBO.0016	Res tag count already reach max value	The maximum number of tags has been reached for the resource.	Delete unnecessary tags.
400	SFS.TURBO.0017	Invalid tag key param	The length of the resource tag key is invalid.	Use valid parameters and try again.
400	SFS.TURBO.0018	Invalid tag value param	The length of the resource tag value is invalid.	Use valid parameters and try again.
404	SFS.TURBO.0019	Invalid Job Id	Invalid job ID.	Use a valid job ID.
400	SFS.TURBO.0020	Invalid flavor	Invalid flavor.	Use a valid flavor.
400	SFS.TURBO.0021	file system not match	Unmatched type. The background disk type is not supported by this file system type.	Ensure that the background disk type is supported by the file system type.
400	SFS.TURBO.0022	backup name already exists	The backup name already exists.	Change the backup name.
400	SFS.TURBO.0023	Invalid flavor ref	Invalid specification code.	Use a valid specification code.
400	SFS.TURBO.0024	Operation is not allowed	Unsupported operation.	Contact technical support.
400	SFS.TURBO.0025	Invalid tag key param	The resource tag key contains invalid characters.	Use valid parameters and try again.

Status Code	Error Code	Error Message	Description	Solution
400	SFS.TURBO.0026	Invalid tag value param	The resource tag value contains invalid characters.	Use valid parameters and try again.
400	SFS.TURBO.0027	Invalid security group	Invalid security group.	Use valid parameters and try again.
400	SFS.TURBO.0028	Invalid crypt key	Invalid KMS key.	Use valid parameters and try again.
400	SFS.TURBO.0029	Subnet has not enough ips	Insufficient IP addresses in the subnet.	Use valid parameters and try again.
400	SFS.TURBO.0030	Ecs resource not enough	The ECS specification is sold out in the selected AZ.	Change the AZ and try again.
400	SFS.TURBO.0031	cache type not exist	The cache type is not found.	Use a valid cache type.
400	SFS.TURBO.0032	EVS Resource Not Enough	Insufficient EVS resources.	Enlarge EVS resources.
500	SFS.TURBO.0033	Get Client Ips Error	Failed to obtain client IP addresses.	Try again. If the fault persists, contact technical support.
400	SFS.TURBO.0034	dedicated storage resource not enough	Insufficient resources in the dedicated storage pool.	Expand the storage pool.
400	SFS.TURBO.0035	The current type does not support backup.	Unsupported type for backup.	Unsupported type for backup.

Status Code	Error Code	Error Message	Description	Solution
500	SFS.TURBO.0036	Failed to obtain the used capacity of the directory	Failed to obtain the used capacity of the directory.	Try again. If the fault persists, contact technical support.
400	SFS.TURBO.0037	Operation conflict, client retry	Operation conflict.	Try again with valid operations.
400	SFS.TURBO.0038	unknown error	Unknown error.	Contact technical support.
400	SFS.TURBO.0039	The VIP quota is insufficient	Insufficient virtual IP address quota.	Apply for a higher quota.
400	SFS.TURBO.0040	Insufficient Security Group Quota	Insufficient security group quota.	Apply for a higher quota.
400	SFS.TURBO.0041	Operation is not allowed	File system version too early.	Contact technical support.
404	SFS.TURBO.0042	Invalid NIC ID	The specified NIC ID is not found or is empty.	Use valid parameters and try again.
400	SFS.TURBO.0100	Invalid file system path	Invalid file system path.	Use a valid file system path.
404	SFS.TURBO.0101	The file system path does not exist	The file system path is not found.	Select a valid file system path.
400	SFS.TURBO.0102	The file system path is not a directory	The file system path is not a directory.	Select a valid file system path.
400	SFS.TURBO.0103	The file system is being processed	The file system is being processed.	Wait until the processing is complete.

Status Code	Error Code	Error Message	Description	Solution
500	SFS.TURBO.0104	Failed to import or export OBS data	Failed to import or export OBS data.	Try again. If the fault persists, contact technical support.
500	SFS.TURBO.0105	Failed to obtain OBS import and export task data	Failed to obtain the OBS import and export data.	Try again. If the fault persists, contact technical support.
400	SFS.TURBO.0106	The OBS task does not exist	The OBS task is not found.	Select an existing OBS task or create an OBS task.
400	SFS.TURBO.0107	OBS protocol error	Backend parameter type is incorrectly configured.	Contact technical support.
400	SFS.TURBO.0108	The OBS endpoint name is incorrect	Incorrect OBS domain name.	Contact technical support.
400	SFS.TURBO.0109	The OBS bucket name is incorrect	Incorrect OBS bucket name.	Use the correct OBS bucket name.
400	SFS.TURBO.0110	OBS agent error	The import or export task failed.	Contact technical support.
400	SFS.TURBO.0111	The OBS configuration list is empty	The OBS configuration list is empty.	Use valid OBS configuration information.

13.3 General Purpose File System Error Codes

If an API calling fails, no result data is returned. You can locate the cause of the error according to the error code of each API. If an API calling fails, HTTP status code 3xx, 4xx or 5xx is returned. The response body contains the specific error code and information. If you are unable to identify the cause of an error, contact

customer service and provide the error code so that we can help you solve the problem as soon as possible.

Error Response Syntax

When an error occurs, the response header information contains:

- Content-Type: application/xml
- HTTP error status code 3xx, 4xx, or 5xx

The response body also contains information about the error. The following is an error response example that shows common elements in the Representational State Transfer (REST) error response body.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>
<Code>NoSuchKey</Code>
<Message>The resource you requested does not exist</Message>
<Resource>/example-filesystem/object</Resource>
<RequestId>001B21A61C6C0000013402C4616D5285</RequestId>
<HostId>RkRCRDJENDc5MzdGQkQ4OUY3MTI4NTQ3NDk2Mjg0M0FB
QUFBQUFBYmJiYmJiYmJD</HostId>
</Error>
```

[Table 13-1](#) describes the meaning of each element.

Table 13-1 Error response elements

Element	Description
Error	Root element that describes the error in an XML response body
Code	HTTP return code that corresponds to the error in the XML response body. For details about error codes, see Table 13-2 .
Message	Error details in the XML response body. For details about error messages, see Table 13-2 .
RequestId	ID of the request whose error response is returned. The ID is used for locating the error.
HostId	ID of the server that returns an error response
Resource	File system or object related to an error.

NOTE

Some error responses contain more detailed information. It is recommended that all error information be logged for easier rectification of errors.

Description

If SFS encounters an error when processing a request, a response containing the error code and description will be returned. [Table 13-2](#) shows the General Purpose File System error codes.

Table 13-2 Error codes

Status Code	Error Code	Error Message	Solution
301 Moved Permanently	PermanentRedirect	The requested file system can be accessed only through the specified address. Send subsequent requests to the address.	Send the request to the returned redirection address.
301 Moved Permanently	WebsiteRedirect	The website request lacks bucketName .	Put the file system name in the request and try again.
307 Moved Temporarily	TemporaryRedirect	Temporary redirection. If the DNS is updated, the request is redirected to the file system.	The system automatically redirects the request or sends the request to the redirection address.
400 Bad Request	BadDigest	The object content MD5 value specified by the client is inconsistent with that received by the system.	Check whether the MD5 value carried in the header is the same as that calculated by the message body.
400 Bad Request	BadDomainName	The domain name is invalid.	Use a valid domain name.
400 Bad Request	BadRequest	Invalid request parameters.	Modify the parameters according to the error details in the message body.
400 Bad Request	IllegalLocationConstraintException	A request without Location is sent for creating a file system in a non-default region.	Send the file system creation request to the default region, or send the request with the Location of a non-default region.
400 Bad Request	InvalidArgument	Invalid parameter.	Modify the parameter according to the error details in the message body.
400 Bad Request	InvalidBucket	The file system to be accessed does not exist.	Try again with another file system name.

Status Code	Error Code	Error Message	Solution
400 Bad Request	InvalidBucketName	The file system name in the request is too long or contains special characters that are not allowed.	Try again with another file system name.
400 Bad Request	InvalidLocationConstraint	The specified Location in the file system creation request is invalid or does not exist.	Correct the Location in the file system creation request.
400 Bad Request	InvalidPolicyDocument	The content of the form does not meet the conditions specified in the policy document.	Modify the policy in the constructed form according to the error details in the message body and try again.
400 Bad Request	InvalidRedirectLocation	Invalid redirect location.	Specifies the correct IP address.
400 Bad Request	InvalidRequest	Invalid request.	Modify the parameter according to the error details in the message body.
400 Bad Request	InvalidRequestBody	The request body is invalid. The request requires a message body but no message body is uploaded.	Upload the message body in the correct format.
400 Bad Request	KeyTooLongError	The provided key is too long.	Use a shorter key.
400 Bad Request	MalformedACLError	The provided XML file is in an incorrect format or does not meet format requirements.	Use the correct XML format to retry.
400 Bad Request	MalformedError	The XML format in the request is incorrect.	Use the correct XML format to retry.
400 Bad Request	MalformedLoggingStatus	The XML format of Logging is incorrect.	Use the correct XML format to retry.
400 Bad Request	MalformedQuotaError	The Quota XML format is incorrect.	Use the correct XML format to retry.
400 Bad Request	MalformedXML	An XML file of a configuration item is in incorrect format.	Use the correct XML format to retry.

Status Code	Error Code	Error Message	Solution
400 Bad Request	MetadataTooLarge	The size of the metadata header has exceeded the upper limit.	Reduce the size of the metadata header.
400 Bad Request	MissingRegion	No region contained in the request and no default region defined in the system.	Carry the region information in the request.
400 Bad Request	MissingRequestBodyError	This error code is returned after you send an empty XML file.	Provide the correct XML file.
400 Bad Request	MissingRequiredHeader	Required headers are missing in the request.	Provide required headers.
400 Bad Request	MissingSecurityHeader	A required header is not provided.	Provide required headers.
400 Bad Request	TooManyBuckets	You have attempted to create more file systems than allowed.	Delete some file systems and try again.
400 Bad Request	TooManyWrongSignatures	The request is rejected due to high-frequency errors.	Replace the Access Key and try again.
400 Bad Request	UnexpectedContent	The request requires a message body which is not carried by the client, or the request does not require a message body but the client carries the message body.	Try again according to the instruction.
400 Bad Request	ContentSHA256Mismatch	The object's SHA-256 value calculated by the client is different from that calculated by the server.	Check whether the SHA-256 value calculated by the client is correct.
403 Forbidden	AccessDenied	Access denied, because the request does not carry a date header or the header format is incorrect.	Provide a correct date header in the request.
403 Forbidden	DeregisterUser	The user has been deregistered.	Top up or register a new account.

Status Code	Error Code	Error Message	Solution
403 Forbidden	InArrearOrInsufficientBalance	The subscriber owes fees or the account balance is insufficient, and the subscriber does not have the permission to perform an operation.	Top up.
403 Forbidden	InvalidAccessKeyId	The access key ID provided by the customer does not exist in the system.	Provide correct access key Id.
403 Forbidden	RequestTimeTooSkewed	There was a large time offset between the OBS server time and the time when the client initiated a request. For security purposes, OBS verifies the time offset between the client and server. If the offset is longer than 15 minutes, the OBS server will reject your requests and this error message is reported.	Check whether there is a large time offset between the client time and server time. If there is, adjust the client time based on your local time (UTC) and try again.
403 Forbidden	SignatureDoesNotMatch	The provided signature does not match the signature calculated by the system.	Check your secret access key and signature algorithm. For details, see Why Don't the Signatures Match?
403 Forbidden	VirtualHostDomainRequired	Virtual hosting access domain name is not used.	Use the virtual hosting access domain name. For details, see Constructing a Request .
403 Forbidden	Unauthorized	The user has not been authenticated in real name.	Authenticate the user's real name and try again.
403 Forbidden	RequestPayerDenied	This is a requester-pays file system.	Carry the header x-obs-request-payer: requester in the access request.

Status Code	Error Code	Error Message	Solution
404 Not Found	NoSuchBucket	The specified file system does not exist.	Create a file system and perform the operation again.
404 Not Found	NoSuchLifecycleConfiguration	The requested lifecycle rule does not exist.	Configure a lifecycle rule first.
405 Method Not Allowed	MethodNotAllowed	The specified method is not allowed against the requested resource. The message "Specified method is not supported." is returned.	The method is not allowed.
408 Request Timeout	RequestTimeout	The socket connection to the server has no read or write operations within the timeout period.	Check the network and try again, or contact technical support.
409 Conflict	BucketAlreadyExists	The requested file system name already exists. The file system namespace is shared by all users.	Try again with another file system name.
409 Conflict	BucketAlreadyOwnedByYou	Your previous request for creating the named file system succeeded and you already own it.	No further action is required.
409 Conflict	BucketNotEmpty	The file system you tried to delete is not empty.	Delete the objects in the file system and then delete the file system.
409 Conflict	ServiceNotSupported	The request method is not supported by the server.	Not supported by the server. Contact technical support.
411 Length Required	MissingContentLength	The HTTP header Content-Length is not provided.	Provide the Content-Length header.
412 Precondition Failed	PreconditionFailed	At least one of the specified preconditions is not met.	Modify according to the condition prompt in the returned message body.

Status Code	Error Code	Error Message	Solution
500 Internal Server Error	InternalError	An internal error occurs. Retry later.	Contact technical support.
501 Not Implemented	ServiceNotImplemented	The request method is not implemented by the server.	Not supported currently. Contact technical support.
503 Service Unavailable	ServiceUnavailable	The server is overloaded or has internal errors.	Try later or contact technical support.
503 Service Unavailable	SlowDown	Too frequent requests.	Reduce your request frequency.

13.4 Obtaining Access Keys (AK/SK)

NOTE

To access SFS using access keys as an IAM user, the programmatic access must be enabled. For details, see [Viewing or Modifying IAM User Information](#).

When calling an API, you need to use the AK/SK to verify the signature. To obtain the AK/SK, perform the following steps:

- Step 1** Log in to the console.
- Step 2** Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list.
- Step 3** On the **My Credentials** page, click **Manage Access Keys**.
- Step 4** In the navigation pane, select **Access Keys**.
- Step 5** Click **Add Access Key**. The **Add Access Key** dialog box is displayed.
- Step 6** Enter the password for login.
- Step 7** Enter the authentication code received in your email or mobile phone.

NOTE

For users created through IAM, if no email address or mobile number is specified during user creation, only the login password needs to be authenticated.

- Step 8** Click **OK** to download the access key file.

NOTE

Keep the AK/SK file properly to prevent information leakage.

----End

13.5 Obtaining a Project ID

Scenarios

A project ID is required for some URLs when an API is called. Therefore, you need to obtain a project ID in advance. Two methods are available:

- [Obtain the Project ID by Calling an API](#)
- [Obtain the Project ID from the Console](#)

Obtain the Project ID by Calling an API

You can obtain a project ID by calling the API used to [query projects based on specified criteria](#).

The API used to obtain a project ID is GET `https://{Endpoint}/v3/projects`. {Endpoint} is the IAM endpoint and can be obtained from [Regions and Endpoints](#). For details about API authentication, see [Authentication](#).

The following is an example response. The value of **id** is the project ID.

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "project_name",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
      },
      "id": "a4a5d4098fb4474fa22cd05f897d6b99",
      "enabled": true
    }
  ],
  "links": {
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects"
  }
}
```

Obtain the Project ID from the Console

To obtain a project ID from the console, perform the following operations:

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list.
On the **API Credentials** page, view the project ID in the project list.

Figure 13-1 Viewing the project ID

