ALIBABA CLOUD

Alibaba Cloud

ApsaraVideo for Media Processing

Developer Guide

Document Version: 20220712

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	Danger: Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	Notice: If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.Overview

ApsaraVideo Media Processing (MPS) offers a cost-efficient, elastic, and scalable solution to transcode media files into different formats that are suitable for playback on various platforms. MPS performs multimodal analysis on the content of media files based on large amounts of data and provides various features such as automated review, intelligent production, and copyright protection. This topic describes the features provided by MPS and how to use them.

Overall architecture

MPS provides the following methods to process media data: using the MPS console, calling API operations, and using SDKs. These methods allow you to use and manage MPS and integrate the transcoding feature into your apps and services.

- Media data processing: MPS provides the transcoding feature that is developed based on the cloud computing services of Alibaba Cloud. The auto scaling feature of the cloud computing services allows you to transcode media data based on your personalized transcoding needs.
- Video AI: The video AI feature can intelligently identify, analyze, review, and understand audio and video content. This helps you improve the click-through rate.
- Workflow: You can create custom workflows in the MPS console or by calling API operations to process audio and video files based on your business requirements.

The following table describes the process of using MPS.

Step	Description
1	A user uploads audio and video files in various forms to the input bucket that the user specifies.
2	MPS processes the audio and video files based on the requirements of the user.
3	MPS stores the processed files in the output bucket that the user specifies.
4	The processed files are transferred to Alibaba Cloud CDN nodes.
5	Alibaba Cloud CDN delivers the processed files to client devices for playback based on the content of the files.

Global configurations

Category	Description	Background information

Category	Description	Background information
Account and access authorization	MPS supports Resource Access Management (RAM) and Security Token Service (STS). You can use one of the following methods to perform authorization operations: • Attach system policies to a RAM user. • Attach custom policies to a RAM user. • Use STS to request a temporary access token for a RAM role.	For each request, MPS authenticates the identity of the user who initiates the request based on the requested operation to check whether the user has the required permissions.
Notification (Optional)	MPS supports notifications by using the following callbacks: • HTTP callbacks (HTTPS compatible) • Alibaba Cloud Message Service (MNS) callbacks	MPS supports notifications sent to MNS queues and topics. To receive callback notifications, you must activate MNS.

Features of MPS

Feature	Description	How to use the feature
Media asset transcoding	Transcodes media data into different formats that are suitable for playback on various platforms.	Use the MPS console or call an API operation.
Video snapshot	Captures snapshots in the JPG format at the specified points in time of a video.	Call an API operation. **Note** You can query but not submit video snapshot jobs in the MPS console.
Video moderation	Accurately identifies illegal content of audio, text, and images in media files. This feature helps you review the content of video files.	Use the MPS console or call an API operation.

Feature	Description	How to use the feature
Media fingerprint	Allows you to extract and compare fingerprint features of images and audio in videos. This feature helps you find duplicate videos, trace the source of video clips, and identify plagiarism.	Use the MPS console or call an API operation.
Video encryption	Encrypts a video to prevent video leaks and hotlinking. This feature is widely used to protect videos in fields such as online education and finance.	Use the MPS console or call an API operation.
Media workflow	Processes uploaded files. For example, you can create a media workflow to transcode, review, package, and analyze uploaded files and generate media files.	Use the MPS console or call an API operation.

2.Concepts

2.1. Job and MPS queue

This topic introduces the basic concepts related to jobs and ApsaraVideo Media Processing (MPS) queues and their associations to help you better understand and use MPS.

Processing logic

1. Submit jobs in synchronous mode: A user submits jobs to be processed in synchronous mode.
2. Submit jobs in asynchronous mode: Jobs are scheduled in an MPS queue based on the job priorities and the sequence in which the jobs are submitted.
3. Send notifications: If a Message Service (MNS) topic or queue is bound to the MPS queue, a notification is sent to the user after each job is run.

Concepts

- Job

  MPS supports multiple types of jobs, such as transcoding jobs, video snapshot jobs, and media information analysis jobs.

  A job contains three key types of information: input files, output files, and parameters. The input files are uploaded for processing, and the output files are generated after the job is run. The parameters are used to set the configurations for implementing specific features.

- MPS queue

  After a user submits jobs in the MPS console or by calling API operations, the jobs wait to be run in an MPS queue based on the job priorities and the sequence in which the jobs are submitted.

  > **Note** You can set different priorities for jobs in an MPS queue. If you submit jobs by calling API operations, you can set priorities that range from 1 to 10 for the jobs. If you submit jobs in the MPS console, the priority for all jobs is automatically set to 6 and cannot be modified. For jobs with the same priority, those that are submitted earlier are run before those that are submitted later. For jobs with different priorities, MPS runs jobs with higher priorities at first.

- Parameter
  - Template parameter

    Different transcoding jobs share some configurations. To reduce the workload of setting duplicate parameters each time you submit a transcoding job, MPS provides you with transcoding templates that combine common parameters required for running transcoding jobs. This simplifies the code required to submit transcoding jobs. Transcoding templates can be shared by different jobs.

  - API parameter

    If you create templates for each combination of parameters, the increased number of templates brings difficulties to template management. Therefore, you can set parameters when you call an API operation in addition to using a template with parameters. The API parameters that you set for a job take effect only for that job.

- Overwriting rule

  If you specify a transcoding template and set API parameters when you call an API operation to submit a transcoding job, the API parameters have a higher priority and overwrite the corresponding parameters in the transcoding template.

  For example, a video file can be transcoded to videos with different definitions, such as the standard definition (SD) and high definition (HD). Videos with different definitions all use the MP4 format and the H.264 video encoding standard. The frame rate for these videos is 25 frames per second. The difference among the videos lies only in the bitrate and resolution. In this case, you can create a template with default parameters that specify the MP4 video format, the H.264 video encoding standard, the frame rate of 25 frames per second, the bitrate of 2 Mbit/s, and the resolution of 1,280 × 720 pixels. If you call an API operation to submit a transcoding job and do not set API parameters, the configurations of the bitrate and resolution specified in the template take effect. If you set the bitrate to 4 Mbit/s and the resolution to 1,920 × 1,080 pixels by setting API parameters, the API parameters take effect.

- Job execution and result query

  The time required for running different types of jobs varies. Some jobs are immediately complete after they are submitted while others are not. In other words, jobs are run in synchronous and asynchronous modes.

  - Synchronous mode

    The results of a job such as a video snapshot job are immediately returned after the job is run.

  - Asynchronous mode

    The results of a job such as a transcoding job can be queried by using periodic polling or notifications.

    - Periodic polling

      Each job is identified by a unique ID, which is immediately returned after the job is submitted in the MPS console or by calling an API operation. You can view the ID of the job in the data that is returned by the MPS console or the API operation. After that, you can query the results of the job by using the job ID. However, you may not obtain the results at the earliest opportunity by using this method.

    - Notification

      If you bind an MNS topic or queue to the MPS queue, you can obtain the results of a job in a timely manner. A notification contains the ID of the job, user data, and results.

      - Job ID

        You can record the ID of each job when you submit the job. Then, you can trace the results of the job based on the job ID sent in a notification.

      - User data

        If you set custom parameters of user data, such as the ID of a commodity, when you submit a job, the custom parameters are returned in a notification. This way, you can associate the user data with your business based on the custom parameters without the need to record the job ID.

2.2. Transcoding template

Different transcoding jobs share some configurations. To reduce the workload of setting duplicate parameters each time you submit a transcoding job, ApsaraVideo Media Processing (MPS) provides you with transcoding templates that combine common parameters required for running transcoding jobs. MPS offers two types of transcoding templates: preset templates and custom templates.

Preset template

Preset templates provide combinations of common parameters required for running transcoding jobs. Preset templates are classified into the following types:

- Static preset template: a transcoding template that can be used without modification. Static preset templates are applicable to scenarios such as audio transcoding, video transcoding, and container format conversion. For example, you can use a static preset template to convert a video file into a high definition (HD) MP4 or 128K MP3 file.
- Narrowband HD™ template: Narrowband HD™ is exclusively available in MPS. Videos that are transcoded by using Narrowband HD™ provide better image quality at the same bitrate. This helps you improve user experience and save costs. MPS provides the preset Narrowband HD™ 1.0 template.
- Intelligent preset template: An intelligent preset template automatically modifies parameters for a transcoding job based on the content of an input file. This reduces the bitrate of the video and saves costs without compromising the image quality.

> **Note** You can use an intelligent preset template only if you submit a transcoding job by calling an API operation. To use an intelligent preset template, you must call the SubmitAnalysisJob operation to submit a media information analysis job. After the media information analysis job is complete, you can call the QueryAnalysisJobList operation to obtain a list of intelligent preset templates that are available to the input file. If the intelligent preset template that you specify for the transcoding job does not exist in the list of valid templates, an error is returned after you submit the transcoding job.

Custom template

If preset templates do not meet your business requirements, you can create a custom template to customize audio, video, container, and transcoding parameters. Each custom template is identified by a unique ID. You can create a regular or Narrowband HD™ template. For more information, see Create a custom transcoding template, Create a Narrowband HD™ 1.0 template.

2.3. Media workflow

A media workflow in ApsaraVideo Media Processing (MPS) contains multiple activities to implement the automatic processing of an input file. This topic introduces the basic concepts and execution rules of media workflows in MPS to help you better understand and use MPS.

Basic concepts

- Media asset

  A media asset contains one input file, such as a video or audio file, and all related output files, such as transcoded files, snapshots, media information, and AI tags. An input file corresponds to only one media asset and is uniquely identified by the ID of the media asset.

- Media Library

Media Library is the collection of all media assets. A media asset is the smallest management unit in Media Library.

- Media workflow

  A media workflow automatically generates media files based on your requirements and is uniquely identified by the ID of the media workflow.

  - Activity

    A media workflow contains multiple nodes. Each node is called an activity. Multiple activities can be run in parallel or one by one. An activity can be an input activity, a notification activity, or a type of job, such as a transcoding job and a video snapshot job.

    > **Note** Features of MPS vary based on different regions. The activities supported by a media workflow are also different among regions. For more information about features supported in different regions, see Regions and endpoints.

    - Input activity

      After you specify the input path for a workflow, the workflow is automatically triggered the moment you upload a media file, such as an audio file or a video file, to the specified input path.

    - Notification activity

      After a workflow is complete, a notification that contains the execution results of the workflow is sent to the specified Message Service (MNS) queue or topic. The execution results include the media ID and the absolute address of the media file. Then, you can determine which media file is processed.

    - Job activity

      All parameters supported by a job can be configured in a job activity.

- Path matching rule

  If the path of an uploaded file matches the input path specified for a workflow, the workflow is automatically triggered. If the prefix of the path of an uploaded file is the same as an input path, the file path matches the input path. For example, the path of an uploaded file is *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video_01.flv*. The following table describes the matching results of multiple Object Storage Service (OSS) input paths.

OSS input path	Matched
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/*	Yes
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C2/*	No
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B*	Yes
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B2/*	No
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/*	Yes
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A2/B/C/*	No
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video*	Yes
*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video_01*	No

- Matching rule of file name extensions

  After you upload a file, the automatic trigger mechanism checks the file name extension to prevent invalid files from being uploaded. For example, a PDF file and a Word file are considered invalid by the automatic trigger mechanism.

  > **Note** If you manually trigger a workflow by calling an API operation, the file name extension is not checked.

  If a file does not have a file name extension or the file name extension matches the extensions in the following lists, the file is considered valid. A file that does not have a file name extension has no separator dot in the file name.

  - File name extensions supported for a video file

    .3gp, .asf, .avi, .dat, .dv, .flv, .f4v, .gif, .m2t, .m3u8, .m4v, .mj2, .mjpeg, .mkv, .mov, .mp4, .mpe, .mpg, .mpeg, .mts, .ogg, .qt, .rm, .rmvb, .swf, .ts, .vob, .wmv, and .webm

  - File name extensions supported for an audio file

    .aac, .ac3, .acm, .amr, .ape, .caf, .flac, .m4a, .mp3, .ra, .wav, .wma, and .aiff

- Workflow execution

  A workflow can be automatically triggered or manually triggered by calling an API operation.

  - Automatic trigger mechanism: A workflow is automatically triggered each time the path of an uploaded media file matches the input path specified for the workflow. If you upload the same file multiple times, the workflow is automatically triggered multiple times. An ID is generated for each execution of the workflow.
  - Manual trigger mechanism: A workflow is triggered each time you call an API operation. The manual trigger mechanism applies to existing video files that are uploaded to OSS buckets but not processed.

- User data

  If you set custom parameters of user data, such as the ID of a commodity, when you trigger a workflow, the custom parameters are returned in a notification. This way, you can associate the user data with your business based on the custom parameters without the need to record the media ID or the absolute path of the media file.

3.Preparations

3.1. Overview

To ensure that all features of ApsaraVideo Media Processing (MPS) work as expected, you must make adequate preparations before you use MPS. This topic describes the required configurations that you must complete before you use MPS.

Preparations

1. Activate MPS.
   -
   -
2. Prepare an Alibaba Cloud account.

   > **Note** The AccessKey pair of an Alibaba Cloud account is the credential for calling Alibaba Cloud APIs and is granted full permissions of the Alibaba Cloud account. Keep the AccessKey pair of your Alibaba Cloud account confidential. To prevent security threats caused by malicious uses, do not expose the AccessKey pair of your Alibaba Cloud account to external channels outside Alibaba Cloud, such as GitHub. We strongly recommend that you use the AccessKey pair of a RAM user instead of your Alibaba Cloud account to call API operations. For more information, see Use RAM to ensure security of your Alibaba Cloud resources.

   Obtain an AccessKey pair to complete identity verification so that you can call server API operations. For more information about how to obtain an AccessKey pair, see Obtain an AccessKey pair.
3. (Optional) Configure a domain name for Content Delivery Network (CDN). We recommend that you perform this operation if you want to use MPS to accelerate the delivery of content on a specified website. A domain name for CDN improves the access speed on the specified website. For more information, see Configure a domain name for CDN.
4. Enable MPS queues as needed. For more information, see Activate or pause an MPS queue. If the MPS queues provided in the MPS console do not meet your requirements, submit a ticket to enable the type of MPS queue that you require.
5. (Optional) Bind a Message Service (MNS) topic or queue to an MPS queue if you want to receive notifications of the MPS queue. For more information, see Enable the feature for sending notifications on transcoding jobs.

3.2. Grant permissions to a RAM user

This topic describes how to authorize a RAM user to access ApsaraVideo Media Processing (MPS) by using an Alibaba Cloud account.

You can log on to the Resource Access Management (RAM) console by using an Alibaba Cloud account and grant permissions to a RAM user. You can grant a RAM user the permissions to access other Alibaba Cloud services. The RAM user can access only the services that are specified by the granted permissions. To allow a RAM user to use MPS, you need to grant the RAM user the permissions to access MPS, Object Storage Service (OSS), Alibaba Cloud CDN, and Message Service (MNS). After you determine the resources of the services that the RAM user needs to access, you can create policies based on policy templates and attach the policies to the RAM user to grant the permissions.

1. Create a RAM user.
   i. Log on to the RAM console.
   ii. In the left-side navigation pane, choose **Identities > Users**. On the page that appears, click **Create User**.
   iii. On the **Create User** page, set the **Logon Name** and **Display Name** parameters in the **User Account Information** section.
   iv. Specify an access method for the RAM user based on your business requirements.

   > **Note** To ensure the security of the Alibaba Cloud account, we recommend that you select only one access method for the RAM user. This prevents the RAM user from using an AccessKey pair to access Alibaba Cloud resources after the RAM user leaves the organization.

Access method	Description
Console Access	If you select this option, you must complete the logon security settings. These settings specify whether to use a system-generated or custom logon password, whether the password must be reset upon the next logon, and whether to enable multi-factor authentication (MFA). **Note** If you select Custom Logon Password for the Console Password parameter, the password that you enter must meet password requirements. For more information about how to configure a password policy, see Configure a password policy for RAM users.
OpenAPI Access	If you select this option, an AccessKey pair is automatically created for the RAM user. The RAM user can call API operations or use SDKs to access Alibaba Cloud resources.

   v. Click **OK**.

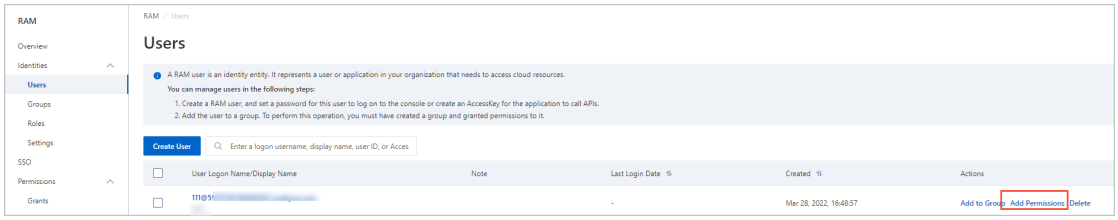
   After you click **OK**, the **Safety Verification** dialog box appears. Complete the verification by following the on-screen instructions. Then, an AccessKey pair is automatically created for the RAM user.

vi. Click **Copy** in the Actions column to copy user information, such as the logon name, logon password, AccessKey ID, and AccessKey secret to the clipboard.

> **Note** We recommend that you store your user information in a secure location for future use.

2. Grant permissions to the RAM user.

i. In the left-side navigation pane, choose Identities > **Users**. On the page that appears, find the created RAM user and click **Add Permissions** in the **Actions** column.



ii. Specify the authorization scope.

Authorization scope	Description
Alibaba Cloud Account	The permissions granted to the RAM user take effect on resources within the current Alibaba Cloud account.
Specific Resource Group	The permissions granted to the RAM user take effect on resources in the specified resource group.

iii. Enter the principal in the **Principal** field.

> **Note** The principal is the RAM user to which you want to grant permissions. By default, the current RAM user is specified. You can also specify another RAM user.

iv. Select the AliyunOSSFullAccess, AliyunCDNFullAccess, AliyunMTSFullAccess, and AliyunMNSFullAccess policies, and click **OK**.

> **Note** The selected policies contain all permissions required to use the MPS, OSS, Alibaba Cloud CDN, and MNS services. To grant fine-grained permissions to the RAM user, go to Step 3 to create a custom policy.

3. (Optional) Create a custom policy.

> **Note** To perform fine-grained control of the permissions granted to RAM users on different services, you can create a custom policy and attach the policy to the RAM user.

i. In the left-side navigation pane, choose **Permissions > Policies**.

ii. On the Policies page, click **Create Policy**. The Create Policy page appears.

iii. Click the JSON tab. Set the Resource field to the Alibaba Cloud Resource Names (ARNs) that you obtain and set the Action field as needed. After that, click Next Step. In the Basic Information section, enter the name of the policy. Click OK and complete the verification.

> **Note** For more information about how to configure a custom policy, see the following policy templates for OSS, Alibaba Cloud CDN, and MNS.

- Policy template for OSS

  This policy grants the following permissions:

  - The permissions to perform all operations on the specified input and output buckets.
  - The permission to view the list of buckets.

```
{
"Version": "1",
"Statement": [
{
"Action": [
"oss:*"
],
"Resource": [
"acs:oss:*:*:$InputBucket",
"acs:oss:*:*:$InputBucket/*",
"acs:oss:*:*:$OutputBucket",
"acs:oss:*:*:$OutputBucket/*"
],
"Effect": "Allow"
},
{
"Action": [
"oss:ListBuckets"
],
"Resource": "*",
"Effect": "Allow"
}
]
}
```

> **Note**
>
> - $InputBucket: the input bucket of MPS. Replace the variable with the name of the bucket to be used as input in the specified workflow.
> - $OutputBucket: the output bucket of MPS. Replace the variable with the name of the bucket to be used as output in the specified workflow.
> - The `oss:ListBuckets` permission is required for a RAM user to perform operations on OSS by using visualization tools. After the permission is granted to the RAM user, the RAM user can query the list of all buckets. However, the RAM user can manage only the input and output buckets specified in the policy. The oss:ListBuckets permission applies only on all buckets, but not a specific bucket.

- Policy template for MNS

  This policy grants the following permissions:

  - The permissions to perform all operations on the specified queues and topics.
  - The permissions to query queues and topics.

```
{
"Version": "1",
"Statement": [
{
"Action": [
"mns:*"
],
"Resource": [
"acs:mns:$Region:$Uid:/queues/$QueueName",
"acs:mns:$Region:$Uid:/topics/$TopicName",
],
"Effect": "Allow"
},
{
"Action": [
"mns:Get*",
"mns:List*"
],
"Resource": "*",
"Effect": "Allow"
}
]
}
```

> **Note**
>
> - $QueueName: the name of the MNS queue. Replace the variable with the name of the queue to be used as a notification destination in the specified workflow.
> - $TopicName: the name of the MNS topic. Replace the variable with the name of the topic to be used as a notification destination in the specified workflow.

- Policy template for Alibaba Cloud CDN

  This policy grants the following permissions:

  - The permissions to perform all operations on the specified domain name for CDN.
  - The permission to query domain names for CDN.

```
{
"Version": "1",
"Statement": [
{
"Action": "cdn:*",
"Resource": [
"acs:cdn:*:$Uid:domain/$DomainName"
],
"Effect": "Allow"
},
{
"Action": "cdn:Describe*",
"Resource": "*",
"Effect": "Allow"
}
]
}
```

> **Note** $DomainName: the domain name for CDN. Replace the variable with the domain name for CDN used in the specified workflow.

- Policy template for RAM

  This policy grants the following permissions:

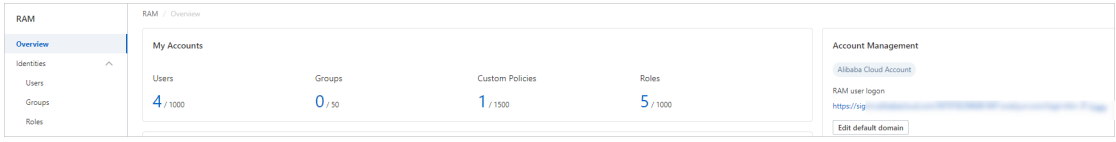
  The permission to query policies attached to RAM roles.

```
{
    "Statement": [{
        "Action": ["ram:ListPoliciesForRole"],
        "Effect": "Allow",
        "Resource": "*"
    }],
    "Version": "1"
}
```

iv. Click **OK**.

4. Log on to the MPS console as the RAM user.

   i. In the left-side navigation pane of the RAM console, click **Overview**. On the Overview page, click the URL under **RAM user logon**.



ii. Set the **Username** and **Password** parameters and click **Log On** to log on to the Alibaba Cloud Management Console.

> **Note** You must reset the password after your first logon as a RAM user.

iii. On the **Product and Service** tab, click **Media Processing Service** to go to the MPS console.

You can use MPS as the RAM user.

3.3. Configure a domain name for CDN

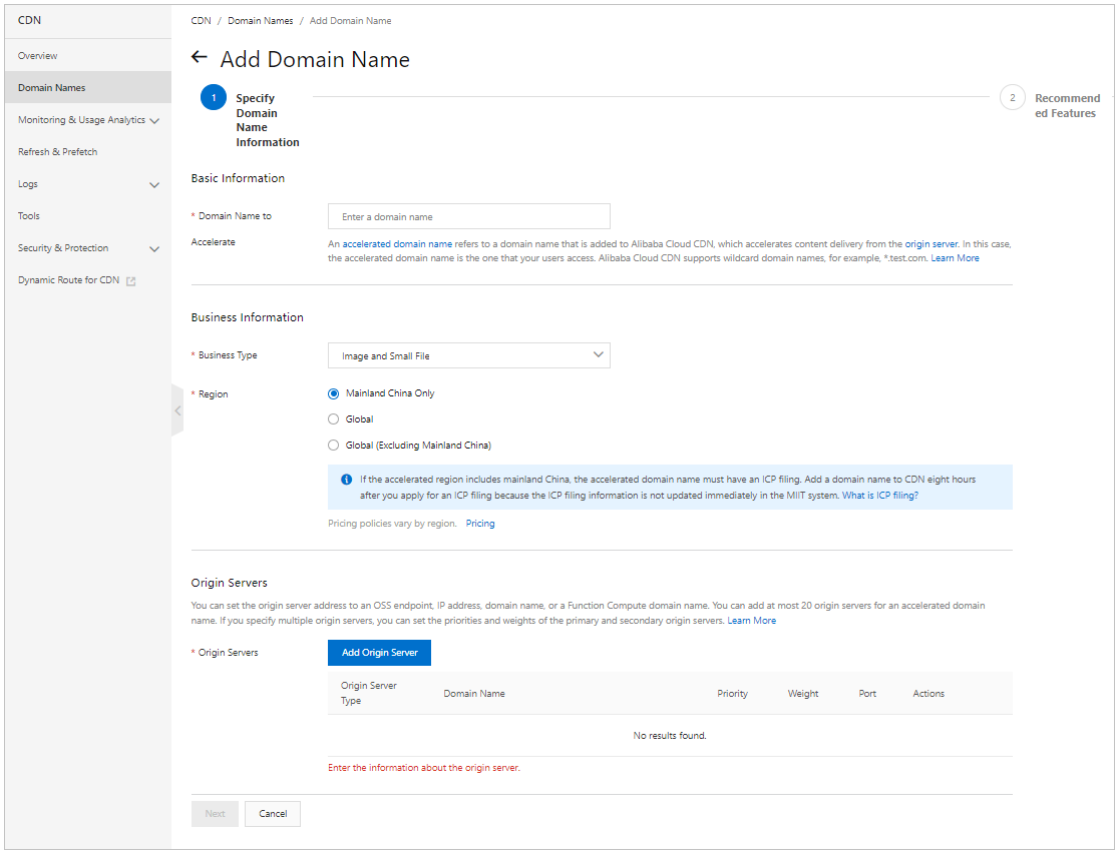
ApsaraVideo Media Processing (MPS) supports accelerated content delivery for resources on a specific website. If you want to accelerate content delivery for resources on a specific website, we recommend that you specify the website as the origin server and add the domain name that you want to accelerate to Alibaba Cloud CDN. You can add a domain name for CDN only in the Alibaba Cloud CDN console. This topic describes how to add a domain name for CDN in the Alibaba Cloud CDN console. Then, you can use this domain name for CDN to accelerate content delivery in MPS.

Prerequisites

Alibaba Cloud CDN is activated. If not, activate this service. For more information, see Activate Alibaba Cloud CDN.

Procedure

1. Log on to the Alibaba Cloud CDN console.
2. In the left-side navigation pane, click **Domain Names**. On the page that appears, click **Add Domain Name**.
3. Configure the domain name.
   i. Set the **Domain Name to Accelerate**, **Business Type**, **Region**, and **Resource Groups** parameters.



Parameter	Description

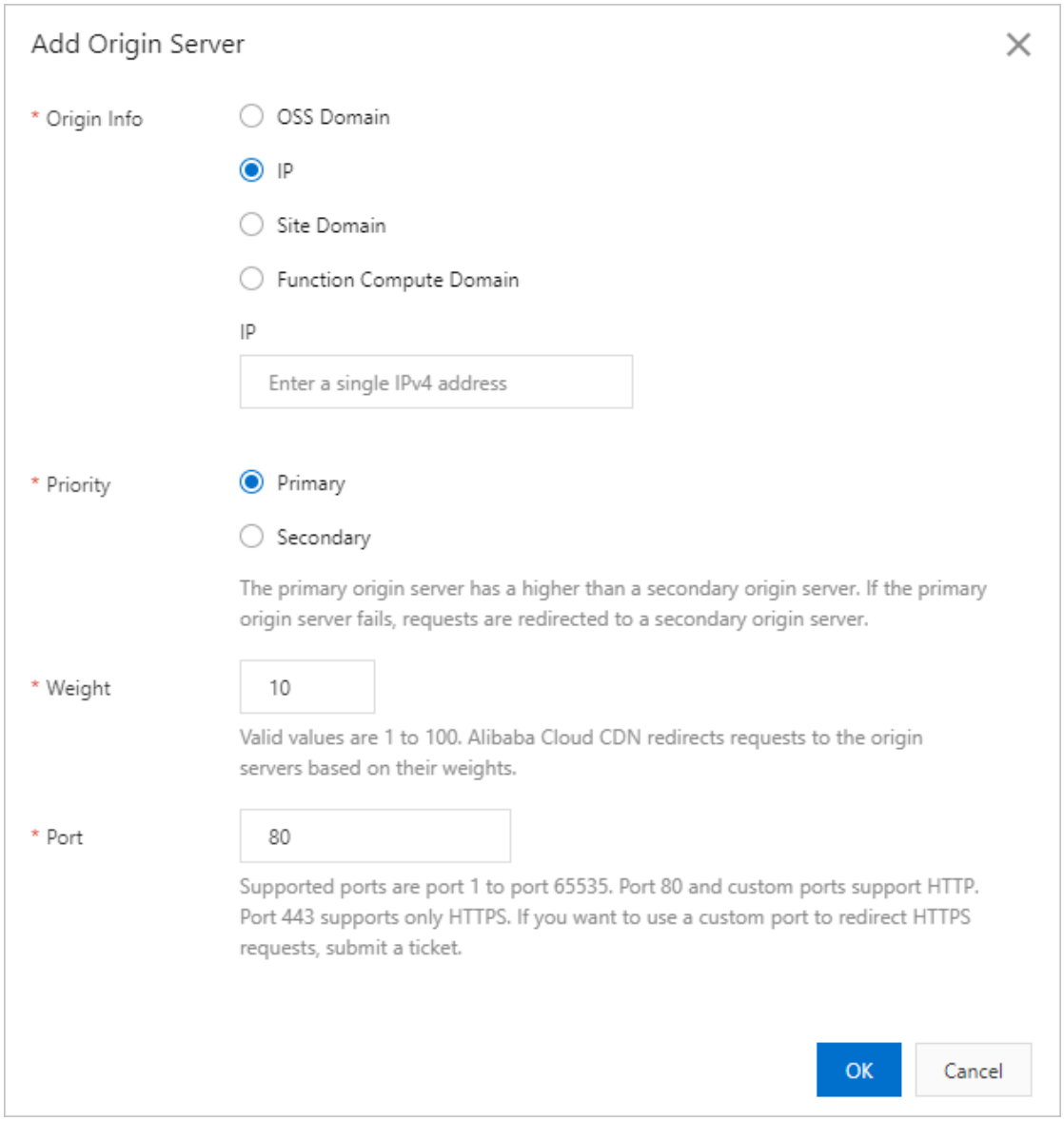
	■ **Domain name type**: The domain name to be accelerated can be a specific domain name such as `example.aliyundoc.com` or a wildcard domain name such as `*.aliyundoc.com`.
	■ **Format**:
	  ■ A domain name must be in lowercase letters such as example.com. Domain names that contain uppercase letters are invalid.
	  ■ If the domain name contains Chinese characters such as aliyun.com, you must apply for an ICP number for the domain name, and use the Punycode tool to convert the domain name to English letters, such as xn--fiq****.xn--eq****. Then, you can specify it as the domain name to be accelerated.

Parameter	Description
Domain Name to Accelerate	■ **Requirements for wildcard domain names:** ■ Alibaba Cloud CDN supports wildcard domain names. For more information about the limits on wildcard domain names, see Does Alibaba Cloud CDN support wildcard domain names? ■ The specified wildcard domain name and the domain names that match the wildcard domain name must belong to the same Alibaba Cloud account. If you need technical support from Alibaba Cloud, submit a ticket. ■ If a wildcard domain name is not added to Alibaba Cloud CDN, you can add domain names that match the wildcard domain name to Alibaba Cloud CDN by using different Alibaba Cloud accounts. ■ Each wildcard domain name can match up to 500 specific domain names. If more than 500 specific domain names are matched, only the first 500 specific domain names can acquire the settings of the wildcard domain name. Other domain names cannot be accelerated by Alibaba Cloud CDN. **Note** The first 500 specific domain names that match the wildcard domain name can be accelerated by Alibaba Cloud CDN. ■ **You cannot add duplicate domain names to Alibaba Cloud CDN.** If the system prompts that the domain name has been added to another Alibaba Cloud service, such as ApsaraVideo VOD or Dynamic Route for CDN (DCDN), you can submit ticket to resolve this issue. ■ **Each Alibaba Cloud account can add a maximum of 50 domain names to Alibaba Cloud CDN.** **Note** However, if the sum of the average daily peak bandwidth values of your domain names exceeds 50 Mbit/s, you can submit a ticket to add more domain names to Alibaba Cloud CDN.

Parameter	Description
	■ **The content delivered from the domain name must be legal and comply with the Terms of Service for Alibaba Cloud CDN.** For more information about the limits, see Limits. ■ **Length**: A domain name cannot exceed 67 characters in length. ■ **ICP filing**: If the accelerated region of a domain name is set to **Global** or **Mainland China Only**, you must apply for an ICP number for the domain name. We recommend that you use Alibaba Cloud ICP Filing System to apply for ICP numbers. ■ **Domain name reclaiming**: If your domain name remains disabled for 120 days, Alibaba Cloud CDN automatically deletes the configuration records that are related to the domain name. This rule also applies to domain names that fail ownership verification. To continue using the domain name, you can log on to the Alibaba Cloud CDN console to add the domain name to Alibaba Cloud CDN again. ■ **Domain name disabling**: For more information, see Rules for disabling accelerated domain names.
Business Type	Select **VOD** from the drop-down list.
Region	■ **Mainland China Only**: All requests are scheduled to edge nodes that are deployed in the Chinese mainland. Requests from regions outside the Chinese mainland are scheduled to edge nodes managed by China Telecom (East China Division). ■ **Global**: All requests are scheduled to the nearest edge nodes. ■ **Global (Excluding Mainland China)**: All requests are scheduled to edge nodes that are deployed in Hong Kong (China), Macao (China), Taiwan (China), and other countries or regions outside the Chinese mainland. However, requests from the Chinese mainland are scheduled to edge nodes that are deployed in Japan, Singapore, and Hong Kong (China).

Parameter	Description
Resource Groups	Select the resource group to which the accelerated domain name belongs from the drop-down list. **Note** If you have never added domain names to Alibaba Cloud CDN, the Resource Groups parameter is not displayed. To use the resource group feature, log on to the Resource Management console, follow the on-screen instructions to enable the resource group feature, and then create a resource group. For more information, see Create a resource group.

ii. Click **Add Origin Server** to add an origin server.



Parameter	Description
Origin Info	Select **OSS Domain**. You can enter the public endpoint that is used to access the bucket over the Internet, such as example-bucket-****.oss-cn-hangzhou.aliyuncs.com. To view the public endpoint of an OSS bucket, log on to the OSS console. You can also select the endpoint of an OSS bucket that belongs to the current Alibaba Cloud account from the Domain Name drop-down list. Internal endpoints of OSS buckets are not supported.

Parameter	Description
Priority	You can set priorities to specify primary and secondary origin servers. The primary origin server has a higher priority than the secondary origin servers. Alibaba Cloud CDN preferably redirects requests to the primary origin server. For example, you specify two origin servers: Origin Server A and Origin Server B. Origin Server A is the primary origin server and Origin Server B is the secondary origin server. In this case, Alibaba Cloud CDN preferably redirects requests to Origin Server A. If Origin Server A fails, Alibaba Cloud CDN redirects user requests to Origin Server B. After Origin Server A recovers, Alibaba Cloud CDN redirects requests to Origin Server A.
Weight	If origin servers have the same priority, Alibaba Cloud CDN redirects requests to the origin servers based on their weights. This balances loads among the origin servers. You can specify a weight based on your business requirements. ■ The weight of an origin server ranges from 1 to 100. An origin server that has a higher weight receives more requests. ■ Default value: 10. For example, you specify Origin Server A and Origin Server B as primary origin servers. If the weight of Origin Server A is 80 and that of Origin Server B is 20, Alibaba Cloud CDN redirects 80% of requests to Origin Server A and 20% of requests to Origin Server B.

Parameter	Description
Port	The port on the origin server that processes requests. The default port is port 80. You can specify a port based on the settings of your origin server. Valid values: 1 to 65535. ■ Default value: 80. ■ If you specify port 443, requests are redirected to the origin server over HTTPS. If you specify port 80 or a custom port, requests are redirected to the origin server over HTTP. **Note** ■ If you want Alibaba Cloud CDN to redirect HTTPS requests to origin servers over custom ports, . ■ If **Origin Protocol Policy** is enabled, custom ports do not take effect. For more information about how to disable the origin protocol policy, see Configure the origin protocol policy. ■ If the origin server is an OSS bucket, OSS determines whether you can specify a custom port.

iii. Click **OK**.

4. Click **Next**. If you have never added domain names to Alibaba Cloud CDN, Alibaba Cloud CDN verifies the ownership of the domain name. For more information, see Verify the ownership of a domain name. If the ownership of the domain name is verified, skip this step.
5. Wait for manual verification. You need to wait for one to two business days before the ownership of a domain name is verified. If the origin server is an Elastic Compute Service (ECS) instance or an OSS bucket, the verification takes a shorter period of time. You can also for urgent needs. After the domain name is verified, you can view the domain name on the **Domain Names** page. If the domain name is in the **Enabled** state, the domain name is added as expected.
6. Configure CNAME record. After the domain name is added to Alibaba Cloud CDN, Alibaba Cloud CDN assigns a CNAME to the domain name. Content delivery acceleration takes effect only after you add the CNAME record for the domain name. For more information, see Add a CNAME record for a domain name.
7. Configure the origin host.
   i. In the left-side navigation pane, click **Domain Names**.
   ii. On the Domain Names page, find the domain name that you want to manage and click **Manage** in the **Actions** column.
   iii. In the left-side navigation pane, click **Back-to-origin**.
   iv. In the **Origin Host** section, click **Modify**.

v. Turn on **Origin Host** and select **Origin Domain** for the **Domain Type** parameter.

> **Note** The origin domain is the domain name of the origin server to which Alibaba Cloud CDN redirects requests, such as `example.com`. If you set the Domain Type parameter to Origin Domain, the domain name of the origin host is specified as the origin domain name, such as `example.com`.

vi. Click **OK**.

8. Enable the video seeking feature.
   i. In the left-side navigation pane, click **Domain Names**.
   ii. On the Domain Names page, find the domain name that you want to manage and click **Manage** in the **Actions** column.
   iii. In the left-side navigation pane, click **Video**.
   iv. In the **Video Seeking** section, turn on **Video Seeking**.

> **Note** After you enable the video seeking feature, Alibaba Cloud CDN supports seeking to a specified position during the playback of an MP4 or FLV file on ApsaraVideo Player for Web. M3U8 files support video seeking even if you have not enabled the video seeking feature in Alibaba Cloud CDN.

After you configure a domain name for CDN, access to the specified website is accelerated.

4.Receive message notifications

4.1. Enable the feature for sending notifications on transcoding jobs

ApsaraVideo Media Processing (MPS) allows you to enable the feature for sending notifications on transcoding jobs to a Message Service (MNS) queue or topic as required.

Enable the feature for sending notifications to an MNS topic

1. Create an MNS topic and a subscription to the MNS topic.
   i. Activate MNS and authorize a RAM user to access MNS. For more information, see Activate MNS and authorize RAM users to access MNS.
   ii. Click **Console** to go to the MNS console.
   iii. In the left-side navigation pane, click **Topics**. On the Topics page, click **Create Topic**.
   iv. In the Create Topic panel, set the parameters that are described in the following table.

> **Note** You are charged a small amount of fees for existing MNS topics every day. Delete unnecessary MNS topics at the earliest opportunity. For more information about billing, see Pricing. Up to 500 messages can be pushed from an MNS topic per second. To push more than 500 messages per second, use Message Queue for Apache RocketMQ.

Parameter	Description
Name	The name of the MNS topic. **Note** The name can be up to 120 characters in length and can contain letters, digits, and hyphens (-). It must start with a letter.
Maximum Message Length	The maximum size of a message that can be sent to the MNS topic. Unit: byte. Valid values: 1024 to 65536. Default value: 65536.
Enable Logging Feature	Specifies whether to enable logging. Valid values: Yes and No. **Note** If logging is enabled, MNS automatically pushes the operation logs of this topic to the specified bucket. You can use the logs to view information such as message traces and message delays.

v. Click **OK**. The MNS topic is created, and the details page of the MNS topic appears.

vi. Click **Create Subscription**.

vii. In the Create Subscription panel, set the parameters that are described in the following table.

Parameter	Description
Name	The name of the subscription. **Note** The name can be up to 255 characters in length and can contain letters, digits, and hyphens (-). It must start with a letter.
Push Type	The push type. Default value: **HTTP**.
Receiver Endpoint	The endpoint of the message receiver. **Note** The endpoint is an HTTP URL, which must start with http:// or https://.
Message Filtering Tag	Optional. The tag that is used to filter messages. **Note** The tag can be up to 16 characters in length.
Retry Policy	The retry policy that is applied when an error occurs during message delivery from the MNS topic to the receiver. Valid values: ■ **Backoff Retry**: retries three times. The retry interval is a random value between 10 and 20 seconds. ■ **Exponential Decay Retry**: retries 176 times within one day at the following retry intervals that are measured in seconds: 2^0, 2^1, ..., 512, 512, ..., and 512.
Message Pushing Format	The format of the message that is pushed to the receiver. ■ **SIMPLIFIED**: The message contains only the published message body and does not contain attribute information. ■ **JSON**: The message is in the JSON format and contains the message body and message attributes. ■ **XML**: The message is in the XML format and contains the message body and message attributes.

viii. Click **OK**. The subscription is created.

2. Enable the feature for sending notifications to the MNS topic for an MPS queue for transcoding.

i. Log on to the MPS console.

ii. In the top navigation bar, select a region.

iii. In the left-side navigation pane, choose **Global Settings > Pipelines**.

iv. On the Pipelines page, find the MPS queue for which you want to enable the notification feature and click **Set Notifications** in the Actions column.

v. In the Notification Settings dialog box, turn on Notifications, select **Topic** for the **Message Type** parameter, and then select the specified MNS topic from the Topic Name drop-down list.

Parameter	Description
Notifications	Turn on Notifications.
Message Type	Select Topic.
Topic Name	Select the specified MNS topic from the Topic Name drop-down list.

vi. Click **OK**. The feature for sending notifications to the specified MNS topic is enabled for the MPS queue.

3. When you create a transcoding job, select the MPS queue. Then, you can receive notifications on the transcoding job.

Enable the feature for sending notifications to an MNS queue

1. Create an MNS queue.
   i. Activate MNS and authorize a RAM user to access MNS. For more information, see Activate MNS and authorize RAM users to access MNS.
   ii. Click **Console** to go to the MNS console.
   iii. In the left-side navigation pane, click **Queues**.
   iv. On the **Queues** page, click **Create Queue**.

v. In the **Create Queue** panel, set the parameters that are described in the following table.

Parameter	Description
Name	The name of the MNS queue.
Maximum Message Length	The maximum size of a message that can be sent to the MNS queue.
Long Polling Period	The maximum period that a ReceiveMessage request can wait till a message is in the MNS queue.
Visibility Timeout Period	The period for which the received message remains in the Inactive state.
Message Retention Period	The maximum period for which a message can be retained in the MNS queue. After the specified period ends, the message is deleted regardless of whether it is consumed.
Scheduled Period	The period after which all messages sent to the MNS queue can be consumed.
Enable Logging Feature	Specifies whether to enable logging.

vi. Click **OK**. The MNS queue is created.

2. Enable the feature for sending notifications to the MNS queue for an MPS queue for transcoding.
   i. Log on to the MPS console.
   ii. In the top navigation bar, select a region.
   iii. In the left-side navigation pane, choose **Global Settings > Pipelines**.
   iv. On the Pipelines page, find the MPS queue for which you want to enable the notification feature and click **Set Notifications** in the Actions column.
   v. In the Notification Settings dialog box, turn on Notifications, select **Queue** for the **Message Type** parameter, and then select the specified MNS queue.

Parameter	Description
Notifications	Turn on Notifications.
Message Type	Select Queue.
Queue Name	Select the specified MNS queue from the Queue Name drop-down list.

   vi. Click **OK**. The feature for sending notifications to the MNS queue is enabled for the MPS queue.
3. When you create a transcoding job, select the MPS queue. Then, you can receive notifications on the transcoding job.

Parameters

The following table describes the parameters in a notification on a transcoding job.

Parameter	Description
jobId	The ID of the job.
type	The type of the job. Valid values: • Transcode: transcoding • Analysis: intelligent template-based analysis • Snapshot: snapshot • MediaInfo: media information
state	The status of the job. Valid values: • Success: The job is successful. • Fail: The job failed.
code	The error code.
msg	The error message.

4.2. Enable the notification feature for a media workflow

If the notification feature is enabled for a media workflow, notifications are sent to the specified Message Service (MNS) queue or topic when the execution of the media workflow starts and ends.

Procedure

1. Create an MNS queue or topic as required and enable the feature for sending notifications to the MNS queue or topic for an ApsaraVideo Media Processing (MPS) queue for transcoding. For more information, see Enable the feature for sending notifications on transcoding jobs.
2. Create a workflow. Configure the Input node as described in the following table. For more information about how to configure other nodes, see Create a workflow.

Parameter	Description
Input Bucket	You do not need to set this parameter. The name of the input bucket automatically appears after you set the Input Path parameter.
Input Path	Click Select next to the Input Path field, set the parameters in the Select Input Path dialog box, and then click OK to specify the input path of the file to be processed.
Encoding Pipeline	Select the MPS queue for transcoding for which you enabled the notification feature in the previous step.

Parameter	Description
Notifications	Turn on Notifications.
Message Type	Select Queue or Topic based on the Message Type parameter that you set when you enabled the notification feature for the MPS queue.
Queue Name or Topic Name	Select the specified MNS queue or topic.

Message format

A message in which a notification on a media workflow is sent is in the JSON format. For more information about the parameters in the message, see the description of media workflow messages in the AddMedia topic.

Message structure:

- Start

  The `activity type` in the `basic attributes of the activity` is `Start`.

- Structure
  - Top level

    A JSON object. Definition: `{Basic attributes of the current activity,Object of workflow execution details}`

    - `Basic attributes of the current activity`

      The basic attributes of the current activity are not an independent object, but key-value pairs that directly belong to the top level. The following code provides an example. Definition: `{ID of the workflow execution instance,Activity name,Activity type,Activity status,Error code and message}`

    - `Object of workflow execution details`

      A JSON object. Definition: `{ID of the workflow execution instance,Media workflow ID,Media workflow name,Media file ID,Input file,Workflow execution type,`**`Array of activity objects`**`,Creation time}`

  - `Array of activity objects`

    A JSON array that contains all activities in the current state. For example, only one Start activity object is included in a message for notifying you that the execution of the workflow starts, and all activity objects are included in a message for notifying you that the execution of the workflow ends. Definition: `[Activity object 1,Activity object 2,…]`

    - `Activity object 1`

      A JSON object. Definition: `{Activity name,Activity type,Job ID,Activity status,Start time,End time,Error code and message}`

    - `Activity object 2`

      The structure is the same as that of `Activity object 1`.

- End

The `activity type` in the `basic attributes of the activity` is `Report`.

- Example


```
{
      "RunId": "8f8aba5a62ab4127ae2add18da20****",
      "Name": "Act-4",
      "Type": "Report",
      "State": "Success",
      "MediaWorkflowExecution": {
        "Name": "ConcurrentSuccess",
        "RunId": "8f8aba5a62ab4127ae2add18da20****",
        "Input": {
            "InputFile": {
                "Bucket": "exampleBucket****",
                "Location": "oss-test",
               "Object": "mediaWorkflow/ConcurrentSuccess/01.wmv",
           }
        },
        "State": "Success",
        "MediaId": "2be491ab4cb6499cd0befe5fcf0c****",
        "ActivityList":
[
{
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-1",
                "Type": "Start",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 41Z",
                "EndTime": "2016-03-15T02: 53: 41Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-2",
                "Type": "Transcode",
                "JobId": "f34b6d1429dd491faa7a6c1c8f90****",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 43Z",
                "EndTime": "2016-03-15T02: 53: 47Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-3",
                "Type": "Snapshot",
                "JobId": "c14150be33304825a5d67cd5364c****",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 44Z",
                "EndTime": "2016-03-15T02: 53: 45Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-4",
                "Type": "Report",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 49Z",
```

```
                "EndTime": "2016-03-15T02: 53: 49Z",
            }
        ],
        "CreationTime": "2016-03-15T02: 53: 39Z",
    }
}
```

4.3. Receive notifications

After a notification is sent, you can receive the notification in the specified Message Service (MNS) queue or topic. This topic describes how to receive notifications.

Prerequisites

The notification feature is enabled. For more information, see Enable the feature for sending notifications on transcoding jobs or Enable the notification feature for a media workflow.

Receive notifications in an MNS queue

- Use MNS SDK for Java to receive notifications.

  > **Note** In this example, MNS SDK for Java is used. For more information about how to use MNS SDKs for other languages to receive notifications, see SDK Reference.

- Use the MNS console to receive notifications.

Receive notifications in an MNS topic

- Use MNS SDK for Java to receive notifications.

  > **Note** In this example, MNS SDK for Java is used. For more information about how to use MNS SDKs for other languages to receive notifications, see SDK Reference.

- Use the MNS console to receive notifications.

5.API path

5.1. Overview

This topic describes how to use the features of ApsaraVideo Media Processing (MPS) by calling API operations.

Features

- Media asset transcoding

  If the files in a workflow cannot meet your requirements in specific scenarios, you can create a transcoding job for the files by calling an API operation. Not all files need to be transcoded, and different videos may require different configurations on transcoding.

- Video merging and cropping

  > **Note** You can use the video merging and cropping features only by calling API operations. You cannot use these features in the MPS console.

  Video merging allows you to merge videos of different formats, bitrates, and resolutions into a longer video of a specified format, bitrate, and resolution. It is usually used to add a start or end part to a video, or to merge the clips of recorded live streams. Video cropping allows you to capture a clip from an original video, and save the clip as a new video. It is usually used to capture highlighted parts from videos.

- Video AI

  The video AI module provides features such as video production, content moderation, media fingerprinting, and smart tagging. You can use these features to identify, analyze, and understand audio and video content.

  - Video production

    The video production feature is implemented based on media AI technologies. It allows you to generate and process media content in different formats. It also provides capabilities such as intelligent thumbnail, intelligent landscape-to-portrait, image matting, figure cutout, intelligent icon blurring, intelligent subtitle removing, subtitle extraction, chorus identification, chorus rhythm identification, and slide video segmentation. All these capabilities improve the efficiency of media content generation and processing.

  - Content moderation

    The content moderation feature is implemented based on large amounts of tagged data and deep learning algorithms. It detects prohibited content (pornography, terrorism, politically sensitive content, and advertisements) from multiple dimensions, such as voice, text, and vision. This feature applies to multiple scenarios, such as content moderation on short video platforms, live streaming platforms, and media platforms.

  - Media fingerprinting

    The media fingerprinting feature is implemented based on binary strings that uniquely identify videos. It allows you to extract and compare the characteristics of images and audio in videos. It helps you find duplicate videos, trace the source of video clips, and identify plagiarism.

- Smart tagging

  The smart tagging feature uses multimodal information fusion and alignment technology to analyze visual information, text, audio, and behavior in videos, which ensures high-accuracy content recognition. It also automatically generates multi-dimensional content tags for recognized content, and therefore converts unstructured information to structured information. This feature applies to scenarios such as media asset queries, personalized recommendations, and intelligent advertising.

- Video encryption

  > **Note** You can use Alibaba Cloud proprietary cryptography to encrypt videos only by using the MPS console. If you create a workflow that applies this encryption method and specify the workflow for a video, MPS automatically starts the workflow after the video is uploaded. You can use HTTP-Live-Streaming encryption to encrypt videos by using the MPS console or by calling an API operation. To use HTTP-Live-Streaming encryption to encrypt a video in the MPS console, you must create a workflow that applies this encryption method and specify the workflow for the video. Therefore, we recommend that you use HTTP-Live-Streaming encryption to encrypt videos by calling an API operation.

  You can encrypt a video to prevent video leak and hotlinking. This method is widely used to protect videos in fields such as online education and finance.

Preparations

1. Activate MPS.
   -
   -
2. Prepare an Alibaba Cloud account.

   > **Note** The AccessKey pair of an Alibaba Cloud account is the credential for calling Alibaba Cloud APIs and is granted full permissions of the Alibaba Cloud account. Keep the AccessKey pair of your Alibaba Cloud account confidential. To prevent security threats caused by malicious uses, do not expose the AccessKey pair of your Alibaba Cloud account to external channels outside Alibaba Cloud, such as GitHub. We strongly recommend that you use the AccessKey pair of a RAM user instead of your Alibaba Cloud account to call API operations. For more information, see Use RAM to ensure security of your Alibaba Cloud resources.

   Obtain an AccessKey pair to complete identity verification so that you can call server API operations. For more information about how to obtain an AccessKey pair, see Obtain an AccessKey pair.
3. Enable MPS queues as needed. For more information, see Activate or pause an MPS queue. If the MPS queues provided in the MPS console do not meet your requirements, submit a ticket to enable the type of MPS queue that you require.
4. (Optional) Bind a Message Service (MNS) topic or queue to an MPS queue if you want to receive notifications of the MPS queue. For more information, see Enable the feature for sending notifications on transcoding jobs.
5. Before you call an API operation to enable a video encryption service, refer to Request syntax to learn about rules on API operations.

Implementation

- Media asset transcoding
- Video merging and cropping
- Video AI
- HLS encryption

5.2. Media asset transcoding

If the transcoding jobs and workflows created in the ApsaraVideo Media Processing (MPS) console cannot meet your business requirements, you can call the SubmitJobs operation to transcode media assets. Set transcoding parameters as required when you call the SubmitJobs operation.

Limits

- Media assets transcoded by calling the SubmitJobs operation support HTTP-Live-Streaming encryption based on the Advanced Encryption Standard 128-bit (AES-128) algorithm, but not Alibaba Cloud proprietary cryptography.
- Media assets transcoded by calling the SubmitJobs operation can be played based on their URLs, but not media IDs.
- To play a media asset on different terminals across different network environments, you can associate different output formats and resolutions with the media ID of the media asset. This way, if you play the media asset based on its media ID, the media asset can be played in different formats and automatically switched between resolutions.

Prerequisites

The following configurations are set before you start a transcoding job:

- The configurations required in the Overview topic are set.
- Media buckets are added in the MPS console. Media assets in MPS are stored in Object Storage Service (OSS) buckets. You must add media buckets in the MPS console before you can specify a bucket as the input or output file path in a job or workflow. For more information, see Add media buckets.
- An MPS queue of the required type is enabled. For more information, see Activate or pause an MPS queue. If no queue of the required type exists in the queue list, submit a ticket to enable a queue of the required type.
- The video to be processed is uploaded to an OSS bucket. For more information, see Video file upload and workflow execution.
- Optional. A custom transcoding template is created if the preset transcoding templates do not meet your business requirements. For more information, see AddTemplate.
- Optional. A watermark template is created if you want to add watermarks to the video to be transcoded. For more information, see AddWaterMarkTemplate.
- Optional. Message Service (MNS) is activated if you want to receive a notification after a transcoding job is complete. For more information, see Enable the feature for sending notifications on transcoding jobs.

Procedure

1. Upload the video to be transcoded to an OSS bucket. You can upload the video by using the MPS console or appropriate OSS tools.

2. Call the SubmitJobs operation to submit a transcoding job.

   For more information, see . Perform the following steps to set transcoding parameters:

   > **Note** Each transcoding job generates an output file. You can submit multiple transcoding jobs at a time.

   i. Select an MPS queue in which the transcoding job is scheduled.

   > **Note** High-speed transcoding jobs must be scheduled in MPS queues. To perform high-speed transcoding, you must enable an exclusive queue for high-speed transcoding. For more information, see Activate or pause an MPS queue.

Scenario	Parameter
High-speed transcoding	Set the PipelineId parameter to the ID of the queue enabled for high-speed transcoding. You can query the queue ID by using the MPS condole or calling the SearchPipeline operation.

   ii. Select a transcoding template as required. If the preset transcoding templates do not meet your business requirements, create a custom transcoding template. For more information, see AddTemplate. The following table describes the parameters that you need to set in different scenarios.

Scenario	Parameter
Narrowband HD™ 1.0 transcoding	Set the TemplateId parameter in output parameters to the ID of a custom Narrowband HD™ 1.0 template or a preset Narrowband HD™ template. You can view the ID of a custom Narrowband HD™ 1.0 template on the **Encoding Template** page. For more information about how to obtain the ID of a preset Narrowband HD™ template, see Preset template details.
Resolution redoubling transcoding	Set the TemplateId parameter in output parameters to one of the following values: ■ S00000003-400040: indicates transcoding from standard definition (SD) to high definition (HD). ■ S00000003-400070: indicates transcoding from 2K to 4K.

   iii. Optional. Select a watermark template.

iv. Optional. Set the Encryption parameter in output parameters as required if you need to encrypt the transcoded video file.

> **Note** The encrypted video file is generated in the M3U8 format. You call an API operation to encrypt a video by using the Base64 algorithm or Key Management Service (KMS). For more information, see Parameter details.

3. After you receive a notification indicating that the transcoding job is complete, call the QueryJobList operation to query the status of the job and obtain the URL of the output file.
4. Optional. To view all the transcoding jobs in the current queue, call the 列出转码作业 operation to obtain the transcoding jobs by job status and job creation time.

5.3. Video merging and cropping

ApsaraVideo Media Processing (MPS) allows you to merge and crop videos by setting parameters when you call the SubmitJobs operation. This topic describes the parameters that you need to set to merge and crop videos. This topic also describes how to set these parameters.

Video merging

Video merging allows you to merge videos of different formats, bitrates, and resolutions into a longer video of the specified format, bitrate, and resolution. Video merging is usually used to add a start or end part to a video, or to merge the clips of recorded live streams.

To merge videos, take note of the following parameters:

> **Note** You can set parameters to merge and crop videos when you call the SubmitJobs operation. For more information, see .

- Input settings

  Specify the videos to be merged and the Object Storage Service (OSS) bucket in which the videos are stored.

  > **Note** The value of the Location parameter, which indicates the region in which the OSS bucket resides, must match the region in which you activate MPS. For example, if the endpoint of your MPS service is `mts.cn-hangzhou.aliyuncs.com`, you must set the Location parameter to `oss-cn-hangzhou`.

- Output settings

  The following table describes the parameters that you can set for the output settings.

Parameter	Description

Parameter	Description
Video	The information about the output file after video merging, such as the width, height, and bitrate. If the aspect ratios of the videos to be merged, including the start and end parts, differ from the aspect ratio of the output file, the system automatically adds black bars. We recommend that you prepare versions of different aspect ratios for the start and end parts to ensure that the output file meets your resolution requirements in different scenarios.
MergeList	The videos to be merged. The sequence of the elements in the video list is the sequence by which the videos are merged. The last element indicates the end part. You can merge up to five videos, including the start and end parts, into an output file. To merge more than five videos, set the MergeConfigUrl parameter. **Note** You can set only one of the MergeList and MergeConfigUrl parameters. Set the following three parameters for each of the videos to be merged: ○ MergeURL The OSS URL of the video to be merged. **Note** All videos to be merged into an output file must reside in the same OSS region as the start part. You cannot merge videos across regions. ○ Start The start point in time of the video to be merged. Set this parameter if you want to merge only part of the video into the output file. Default value: 0. ○ Duration The length of the video to be merged, relative to the start point in time specified by the Start parameter. Set this parameter if you want to merge only part of the video into the output file. By default, the length is the period from the start point in time specified by the Start parameter to the end of the video.

Parameter	Description
MergeConfigUrl	The OSS URL of the configuration file for the video merging job. The content in the configuration file is in the JSON format. The file content is the same as the value of the MergeList parameter. **Note** ◦ You can set only one of the MergeList and MergeConfigUrl parameters. ◦ The sequence of the elements in the video list is the sequence by which the videos are merged. The last element indicates the end part. You can merge up to 100 videos, including the start and end parts, into an output file.

Video cropping

To crop videos, take note of the following parameters:

- Input settings

  Specify one or more videos to be cropped and the OSS bucket in which the videos are stored.

  Note The value of the Location parameter, which indicates the region in which the OSS bucket resides, must match the region in which you activate MPS. For example, if the endpoint of your MPS service is `mts.cn-hangzhou.aliyuncs.com`, you must set the Location parameter to `oss-cn-hangzhou`.

- Output settings

  The following table describes the parameters that you can set for the output settings.

Parameter	Description

Parameter	Description
TimeSpan	The time span of the clip to be captured from the original video. Set this parameter based on your business requirements. The TimeSpan parameter contains the following three parameters: ○ Seek: the start point in time of the clip to be captured. ○ Duration: the length of the clip to be captured. ○ End: the length of the end part of the original video to be cropped out. Note If you set the End parameter, the Duration parameter becomes invalid.
ConfigToClipFirstPart	Specifies whether to crop the first part of the videos to be merged. A value of true indicates that the first part of the videos is cropped before the videos are merged. A value of false indicates that the videos are merged into an output file, which is then cropped. Default value: false.

Sample code

In this example, the resolution of the original video is 1,280 × 720 pixels, the start and end parts to be added are two MP4 videos of 640 × 480 pixels, and the resolution of the output video is 1,280 × 720 pixels. Therefore, in the output video, the start and end parts have black bars on the left and right sides.

The following table describes the topics in which you can obtain the sample code of different languages.

Language	References
Java	Sample code of MPS SDK for Java
Python	Sample code of MPS SDK for Python
PHP	Sample code of MPS SDK for PHP

FAQ

If the audio of a merged video cannot be played on some iOS devices, the encoding profile of the merged video is not suitable for the definitions supported by these iOS devices due to the unsatisfactory decoding capability of iOS. In this case, when you call the SubmitJobs operation, you can set the encoding profile of the output file to main. This ensures that the output file is suitable for standard-definition devices.

5.4. Video AI

ApsaraVideo Media Processing (MPS) provides video AI features such as intelligent video production, content moderation, media fingerprinting, and smart tagging. This topic describes how to use these features by calling API operations.

Prerequisites

The following configurations are set before you start various video AI jobs:

- The configurations required in the Overview topic are set.
- Media buckets are added in the MPS console. Media assets in MPS are stored in Object Storage Service (OSS) buckets. You must add media buckets in the MPS console before you can specify a bucket as the input or output file path in a job or workflow. For more information, see Add media buckets.
- An MPS queue of the required type is enabled. For more information, see Activate or pause an MPS queue. If no queue of the required type exists in the queue list, submit a ticket to enable a queue of the required type.
- The video to be processed is uploaded to an OSS bucket. For more information, see Video file upload and workflow execution.
- Optional. Message Service (MNS) is activated if you want to receive a notification after a video AI job is complete. For more information, see Enable the feature for sending notifications on transcoding jobs.

Content moderation

The content moderation feature is implemented based on large amounts of tagged data and deep learning algorithms. It detects prohibited content (pornography, terrorism, politically sensitive content, and advertisements) from multiple dimensions, such as voice, text, and vision. This feature applies to multiple scenarios, such as content moderation on short video platforms, live streaming platforms, and media platforms.

1. Submit a content moderation job.

   For videos, images, and audio files that are uploaded to an OSS bucket, you can call the SubmitMediaCensorJob operation to submit a content moderation job.

   > **Note** You can set different moderation policies for different types of prohibited content. For example, you can implement a high detection accuracy rate and recall rate for pornographic content and specify detection scenarios for terrorism-related and politically sensitive content. To adjust moderation policies, submit a ticket to provide the unique ID (UID) of your Alibaba Cloud account, region, and business requirements.

2. Receive a callback notification.

   If you activate MNS and configure a queue or topic to receive notifications, MPS sends a notification to the queue or topic after the content moderation job is complete. For more information, see Receive notifications.

3. Query the results of the content moderation job.

   You can call the QueryMediaCensorJobDetail operation to query the status and results of the content moderation job by job ID.

Media fingerprinting

The media fingerprinting feature is implemented based on binary strings that uniquely identify videos. It allows you to extract and compare the characteristics of images and audio in videos. It helps you find duplicate videos, trace the source of video clips, and identify plagiarism.

1. Create a media fingerprint library.

   Submit a job to create a media fingerprint library. The job is executed to return information about the new library. The new library enters the active state after it is created. When you submit the job, provide the UID of your Alibaba Cloud account. After the media fingerprint library is created, bind the UID of your Alibaba Cloud account to the media fingerprint library.

   > **Note** A media fingerprint library is used to record and store the media fingerprints of videos. Take note of the following items when you create a media fingerprint library:
   >
   > - You must use the UID of your Alibaba Cloud account to create a media fingerprint library.
   > - Multiple Alibaba Cloud accounts can share the same media fingerprint library.
   > - You can use the UID of your Alibaba Cloud account to create multiple media fingerprint libraries based on your business requirements.

2. Submit a job to extract and compare the media fingerprints of new videos against the media fingerprint library.
   - If the total length of your existing videos does not exceed 1 million minutes, you can call the SubmitFpShotJob operation to submit a job to extract and compare the media fingerprints of existing videos.
   - If incremental videos are added in real time, you can also call the SubmitFpShotJob operation to compare the media fingerprints of incremental videos. You can make media fingerprint comparisons for incremental and existing videos in one job. Alternatively, you can call the SubmitFpShotJob operation to submit a media fingerprinting job after existing videos are imported.
3. Query the results of the media fingerprinting job.

   To query the media fingerprint comparison results of incremental videos, call the QueryFpShotJobList or ReportFpShotJobResult operation.

   The media fingerprint comparison results contain the following information:
   - Whether duplicate videos are found
   - The information about duplicate videos
   - The interval of the timeline of a video during which the video content shows a duplication from another video
   - The degree of duplication between two videos, which is specified by a number from 0 to 1

5.5. HLS encryption

The video encryption feature allows you to encrypt a video. This feature is widely used to prevent video leaks and hotlinking in scenarios such as online education and finance. ApsaraVideo Media Processing (MPS) supports two encryption methods: Alibaba Cloud proprietary cryptography and HTTP-Live-Streaming (HLS) encryption. We recommend that you use the former method in the MPS console. This topic describes how HLS encryption works and how to use HLS encryption by calling API operations.

How it works

MPS uses the envelope encryption technology to encrypt videos. You can use Alibaba Cloud Key Management Service (KMS) to generate a data key (DK) and an enveloped data key (EDK). Then, you can use the DK to encrypt a video, and store the encrypted video together with the EDK. If you want to play the video, the player uses a decryption service to obtain the DK and thereby decrypt the video.

> **Note**
> - A DK is also called a plaintext key. It is used for video encryption.
> - An EDK is also called a ciphertext key. It is a ciphertext data key encrypted by using the envelope encryption technology. It is used to decrypt a DK and obtain the plaintext data key.
> - HLS encryption requires you to preserve your DKs.

1. Activate the following services: MPS, Object Storage Service (OSS), Resource Access Management (RAM), KMS, and Alibaba Cloud Content Delivery Network (CDN).

   > **Note**
   > - OSS is a data storage service provided by Alibaba Cloud. Media files for jobs of MPS are stored in OSS buckets.
   > - RAM is an access management service provided by Alibaba Cloud. You can use RAM to grant KMS access permissions to MPS.
   > - KMS is a security management service provided by Alibaba Cloud. You can use KMS to generate DKs, encrypt files, and decrypt files.
   > - Alibaba Cloud CDN is a content delivery network provided by Alibaba Cloud. In the process of HLS encryption, Alibaba Cloud CDN dynamically modifies the decryption uniform resource identifier (URI) in the M3U8 file and returns the decryption URI to the player.

2. Grant KMS access permissions to MPS.

   > **Note** This ensures that MPS can call the GenerateDataKey operation of KMS to generate DKs and EDKs during video encryption.

3. Configure a domain name for CDN for the OSS bucket that stores output files of MPS jobs. Add a CNAME record and configure the origin host for the domain name for the CDN.
4. Create a workflow for video encryption and specify information such as the OSS bucket that stores output files and the key URI.

   The key URI specifies the endpoint of your services. The data about the key URI is contained in the M3U8 file generated by MPS after video encryption.
5. Upload the video to be encrypted and specify the created workflow for the video.
6. After the video is uploaded, MPS automatically triggers the workflow.

   Then, MPS calls the GenerateDataKey operation to generate a DK and an EDK, and uses the DK to encrypt the video. After the video is encrypted, MPS writes the data about the key URI and EDK to the M3U8 file.

7. MPS stores the M3U8 file and the TS file in the OSS bucket that stores output files.

1. Construct a token issuance service to generate the MtsHlsUriToken parameter.

   > **Notice** The token issuance service is used to generate the MtsHlsUriToken parameter.

2. Call the KMS decryption operation to construct a decryption service and return the DK to the player.

   > **Notice** After you call a KMS operation to receive the Base64-encoded DK from KMS, you must decode the DK by using the Base64 algorithm and return the decoded DK to the player.

3. Call the QueryMediaList operation of MPS to query the OSS URL of the M3U8 file, add the MtsHlsUriToken parameter to the OSS URL, and then return the OSS URL to the player.
4. The player uses the MtsHlsUriToken parameter and the DK to require the streaming URL of the video from Alibaba Cloud CDN. Then, Alibaba Cloud CDN modifies the M3U8 file and returns the key URI and EDK to the player. Then, the player decrypts and plays the video.

Code logic

In the procedures for HLS encryption and decryption, you must implement the following code logic:

- Create a workflow for video encryption.

  > **Note** Although you can create a workflow in the MPS console, we recommend that you create a workflow by using a server SDK.

- Construct a token issuance service to generate the MtsHlsUriToken parameter, which is used as the decryption token. Verify the validity of the decryption token. We recommend that you use each decryption token only once.
- Call the KMS decryption operation to construct a decryption service. Decode the DK by using the Base64 algorithm and return the decoded DK to the player.

Preparations

Before you use HLS encryption in MPS, make the following preparations:

1. Activate relevant Alibaba Cloud services, including MPS, OSS, KMS, RAM, and Alibaba Cloud CDN.

   If you have not activated these services, perform the following steps:

   i. Activate MPS. For more information, see Activate MPS.

   ii. Activate OSS. For more information, see Activate OSS.

   iii. Activate KMS. For more information, see Activate KMS.

   iv. Activate RAM and grant required permissions. For more information, see Create a RAM user and grant permissions to the RAM user.

   v. Activate Alibaba Cloud CDN. For more information, see Activate Alibaba Cloud CDN.

2. Grant KMS access permissions to MPS.

   i. Log on to the RAM console.

   ii. Click **Authorize** in the left-side navigation pane. On the Grants page, click Grant Permission to go to the **Authorize** page.

iii. Enter *AliyunMTSDefaultRole* in the **Principal** search box. Select a role that is created by the system and can be used in MPS.

iv. Enter *KMS* in the search box in the **Select Policy** section. Select *AliyunKMSFullAccess*, and click **OK**.

Then, MPS are granted the permissions to access KMS. After MPS receives a video encryption request, MPS can call a KMS operation to obtain the DK.

3. Configure a domain name for CDN for the OSS bucket that stores output files, and configure the origin host for the domain name for CDN. For more information, see Configure a domain name for CDN. If the domain name for CDN and the origin host are configured, skip this step.

> **Note** You can enter the public domain name of the OSS bucket, such as `exampleBucket****.oss-cn-hangzhou.aliyuncs.com`. You can obtain the public domain name in the OSS console. Alternatively, you can select the OSS bucket that stores output files and requires content delivery acceleration within the same Alibaba Cloud account. Internal domain names of OSS buckets are not supported.

Encrypt a video

To encrypt a video, perform the following operations:

1. Create a workflow for video encryption.

   To create a workflow of this type, you must use an Alibaba Cloud SDK, and add MPS dependencies. You can view the sample code of workflow creation based on the programming language you use. For more information, see the following table.

   > **Notice** When you create a workflow, you must provide the key URI. During video encryption, MPS writes the key URI to the M3U8 file and stores the file in the OSS bucket that stores output files. An example of the key URI is `example.aliyundoc.com`.

Programming language	SDK	Sample code
Java	SDK for Java	Create HLS standard encryption workflow
Python	SDK for Python	Create HLS standard encryption workflow
PHP	SDK for PHP	Create a workflow for HLS encryption

2. Upload a video to trigger the workflow. You can upload a video by using the MPS console or OSS console. For more information, see Upload a video.

   > **Note** If you specify the created workflow for the video in the upload configuration, after the video is uploaded, MPS automatically triggers the workflow.

   After the video is encrypted, log on to the OSS console and view the M3U8 file in the OSS bucket that stores output files. The following code provides an example of the M3U8 file:

```
#EXTM3U
     #EXT-X-VERSION:3
     #EXT-X-TARGETDURATION:5
     #EXT-X-MEDIA-SEQUENCE:0
     #EXT-X-KEY:METHOD=AES-128,URI="https://example.aliyundoc.com?Ciphertext=aabbccdd
eeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****"
     #EXTINF:4.127544,
     15029611683170-00001.ts
     #EXT-X-ENDLIST
```

In this example, the key URI you configure and the EDK are contained in the URI field.

Play an HLS-encrypted video

To play an HLS-encrypted video, perform the following operations:

1. Construct a token issuance service.

   > **Note** You must construct the token issuance service based on your encryption method to ensure a high security level.

2. Construct a decryption service.

   Construct a local HTTP service to decrypt the video and obtain the decryption key. The following sample code provides examples on how to construct the decryption service in Java and Python.

   - Sample code in Java

     Dependencies required by the SDK for Java:

     - Java SDK Core
     - Java SDK KMS

     Sample code in Java:

```
package com.aliyun.smallcode;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.ProtocolType;
import com.aliyuncs.kms.model.v20160120.DecryptRequest;
import com.aliyuncs.kms.model.v20160120.DecryptResponse;
import com.aliyuncs.profile.DefaultProfile;
import com.sun.net.httpserver.Headers;
import com.sun.net.httpserver.HttpExchange;
import com.sun.net.httpserver.HttpHandler;
import com.sun.net.httpserver.HttpServer;
import com.sun.net.httpserver.spi.HttpServerProvider;
import org.apache.commons.codec.binary.Base64;
import java.io.IOException;
import java.io.OutputStream;
import java.net.HttpURLConnection;
import java.net.InetSocketAddress;
import java.net.URI;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
public class AuthorizationServer {
```

```
private static DefaultAcsClient client;
static {
String region = "";
String accessKeyId = "";
String accessKeySecret = "";
client = new DefaultAcsClient(DefaultProfile.getProfile(region, accessKeyId, accessKe
ySecret));
}
public class AuthorizationHandler implements HttpHandler {
public void handle(HttpExchange httpExchange) throws IOException {
String requestMethod = httpExchange.getRequestMethod();
if(requestMethod.equalsIgnoreCase("GET")){
// Obtain the EDK from the video URL.
String ciphertext = getCiphertext(httpExchange);
if (null == ciphertext)
return;
// Decrypt the DK obtained from KMS and decode the DK by using the Base64 algorithm.
byte[] key = decrypt(ciphertext);
// Configure the headers.
setHeader(httpExchange, key);
// Return the decoded DK.
OutputStream responseBody = httpExchange.getResponseBody();
responseBody.write(key);
responseBody.close();
}
}
private void setHeader(HttpExchange httpExchange, byte[] key) throws IOException {
Headers responseHeaders = httpExchange.getResponseHeaders();
responseHeaders.set("Access-Control-Allow-Origin", "*");
httpExchange.sendResponseHeaders(HttpURLConnection.HTTP_OK, key.length);
}
private byte[] decrypt(String ciphertext) {
DecryptRequest request = new DecryptRequest();
request.setCiphertextBlob(ciphertext);
request.setProtocol(ProtocolType.HTTPS);
try {
DecryptResponse response = client.getAcsResponse(request);
String plaintext = response.getPlaintext();
// Note: You must decode the DK by using the Base64 algorithm.
return Base64.decodeBase64(plaintext);
} catch (ClientException e) {
e.printStackTrace();
return null;
}
}
private String getCiphertext(HttpExchange httpExchange) {
URI uri = httpExchange.getRequestURI();
String queryString = uri.getQuery();
String pattern = "Ciphertext=(\\w*)";
Pattern r = Pattern.compile(pattern);
Matcher m = r.matcher(queryString);
if (m.find())
return m.group(1);
else {
```

```
System.out.println("Not Found Ciphertext");
return null;
}
}
}
private void startService() throws IOException {
HttpServerProvider provider = HttpServerProvider.provider();
// Configure a listener on port 8888, which can accept 10 requests at a time.
HttpServer httpserver = provider.createHttpServer(new InetSocketAddress(8888), 10);
httpserver.createContext("/", new AuthorizationHandler());
httpserver.start();
System.out.println("server started");
}
public static void main(String[] args) throws IOException {
AuthorizationServer server = new AuthorizationServer();
server.startService();
}
}
```

- Sample code in Python

  Dependencies required by the SDK for Python:

  - pip install aliyun-python-sdk-core
  - pip install aliyun-python-sdk-kms
  - pip install aliyun-python-sdk-mts

  Sample code in Python:

```
# -*- coding: UTF-8 -*-
from BaseHTTPServer import BaseHTTPRequestHandler
from aliyunsdkcore.client import AcsClient
from aliyunsdkkms.request.v20160120 import DecryptRequest
import cgi
import json
import base64
import urlparse
client = AcsClient("","","");
class AuthorizationHandler(BaseHTTPRequestHandler):
def do_GET(self):
self.check()
self.set_header()
cipertext = self.get_cihpertext()
plaintext = self.decrypt_cihpertext(cipertext)
print plaintext
key = base64.b64decode(plaintext)
print key
self.wfile.write(key)
def do_POST(self):
pass
def check(self):
#check MtsHlsUriToken, etc.
pass
def set_header(self):
self.send_response(200)
#cors
self.send_header('Access-Control-Allow-Origin', '*')
self.end_headers()
def get_cihpertext(self):
path = urlparse.urlparse(self.path)
query = urlparse.parse_qs(path.query)
return query.get('Ciphertext')[0]
def decrypt_cihpertext(self, cipertext):
request = DecryptRequest.DecryptRequest()
request.set_CiphertextBlob(cipertext)
response = client.do_action_with_exception(request)
jsonResp = json.loads(response)
return jsonResp["Plaintext"]
if __name__ == '__main__':
# Start a simple server, and loop forever
from BaseHTTPServer import HTTPServer
print "Starting server, use  to stop"
server = HTTPServer(('127.0.0.1', 8888), AuthorizationHandler)
server.serve_forever()
```

3. Call the QueryMediaList operation of MPS to query the streaming URL of the video.

   You can call the operation in OpenAPI Explorer or by integrating an SDK.

4. Play the encrypted video.

   You can use ApsaraVideo Player or a third-party player to play the encrypted video.

   - If you use a third-party player, specify the playback logic.

- If you use ApsaraVideo Player, obtain the security token and authentication information as required before the playback. For more information, see Play videos.

  You can also use an online player to test the playback of the video encrypted by HLS encryption.

  For example, you can use the ApsaraVideo Player diagnosis tool. When you use this tool, enter the streaming URL of the video in the Source field and click **Play**.

  > **Note** On the browser debugging page, you can see that the player sends a request to the authentication server, obtains the decryption key, and then decrypts and plays the video.

  The following procedure describes how ApsaraVideo Player tests the playback:

  - After ApsaraVideo Player receives the streaming URL, it replaces the domain name of the OSS bucket with a domain name for CDN. Then, ApsaraVideo Player adds the MtsHlsUriToken parameter, which is used as the decryption token, to the domain name for CDN, and sends a request to Alibaba Cloud CDN for a modified streaming URL. Sample request: `https://example.aliyundoc.com/test_01.m3u8?MediaId=fbbf98691ea44b7c82dd75c5bc8b****&MtsHlsUriToken=<Token>`.

    > **Notice**
    >
    > The MtsHlsUriToken parameter is automatically added if you use ApsaraVideo Player. If you use other players, you must manually add the MtsHlsUriToken parameter.

  - After Alibaba Cloud CDN receives the request, it dynamically modifies the key URI in the M3U8 file and returns the modified streaming URL to the player. For example, if the original streaming URL is `https://example.aliyundoc.com?Ciphertext=aabbccddeeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****`, the returned streaming URL is `https://example.aliyundoc.com?Ciphertext=aabbccddeeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****&MtsHlsUriToken=<Token>`.
  - The player parses and accesses the URI in the EXT-X-KEY tag of the M3U8 file to obtain the decryption key. Call the Decrypt operation of KMS, decode the obtained DK by using the Base64 algorithm, and then return the decoded DK to the player. The player uses the DK to decrypt the TS file and plays the video.

5.6. Video file upload

5.6.1. Create a RAM user and grant permissions to the RAM user

This topic describes how to create a RAM user and grant permissions to the RAM user before you decrypt and play a video.

1. Create a RAM user.
   i. Log on to the RAM console.
   ii. In the left-side navigation pane, choose **Identities > Users**. On the page that appears, click **Create User**.

iii. On the **Create User** page, set the **Logon Name** and **Display Name** parameters in the **User Account Information** section.

iv. Select **Console Access** and **OpenAPI Access** for the **Access Mode** parameter. Set the **Console Password**, **Password Reset**, and **Multi-factor Authentication** parameters as required. This way, you can create a RAM user that has the same permissions of ApsaraVideo Media Processing (MPS) as your Alibaba Cloud account.

v. Click **OK**. The Safety Verification dialog box appears. After the verification, the system automatically generates the AccessKey ID and AccessKey secret of the RAM user.

vi. You can click Copy in the Actions column to copy the AccessKey ID and AccessKey secret. Make sure that they are properly kept.

> **Notice** MPS uses symmetric encryption based on an AccessKey pair to verify the identity of a requester. Make sure that you properly keep your AccessKey pair.
>
> - The AccessKey ID is used to identify a user.
> - The AccessKey secret is used to encrypt and verify signature strings. You must keep your AccessKey secret confidential.
> - An AccessKey pair consists of an AccessKey ID and an AccessKey secret.

2. Create a role.

i. In the left-side navigation pane, choose **Identities > Roles**.

ii. On the **Roles** page, click **Create Role**.

iii. In the **Select Role Type** step of the Create Role panel, set the **Select Trusted Entity** parameter to **Alibaba Cloud Account**. Click **Next**.

iv. In the Configure Role step, enter a RAM role name in the **RAM Role Name** field. Set the **Select Trusted Alibaba Cloud Account** parameter to **Current Alibaba Cloud Account** and click **OK**.

v. Click **Close** in the lower-left corner of the Finish step.

vi. Find the created RAM role in the RAM role list. Click the role name to go to the role details page. In the **Basic Information** section, copy the value of the **ARN** parameter and properly keep it for follow-up operations.

3. Grant permissions to the RAM role.

i. On the **Roles** page, find the created RAM role and click **Add Permissions** in the **Actions** column.

ii. In the Add Permissions panel, set the **Authorized Scope** parameter to **Alibaba Cloud Account**.

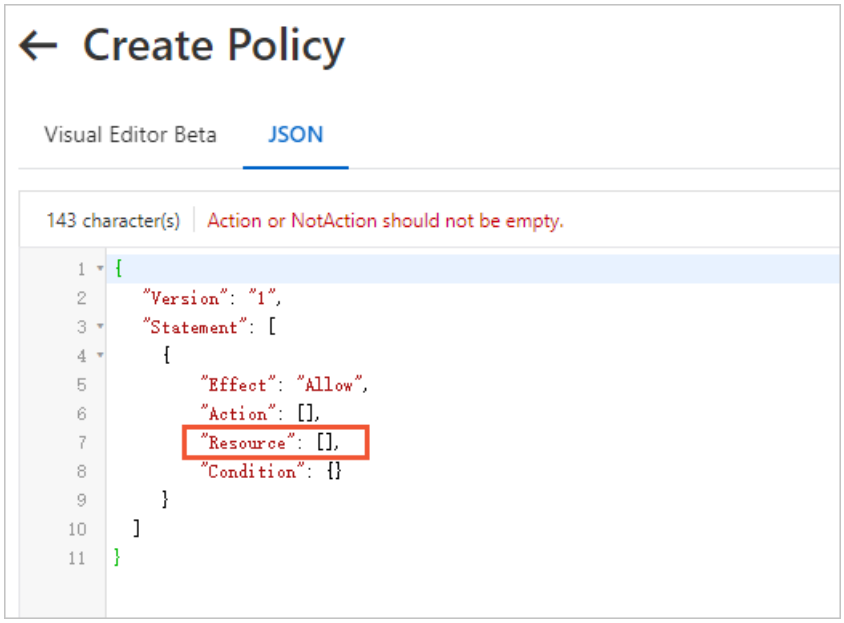
Value	Description
Alibaba Cloud Account	The permissions granted to the RAM role take effect on resources within the current Alibaba Cloud account.
Specific Resource Group	The permissions granted to the RAM role take effect on resources in the specified resource group.

iii. In the **Principal** field, the system automatically enters the name of the current RAM role.

iv. In the **Select Policy** section, click **System Policy**, select one or more required policies in the policy list, and then click **OK**. The policies are attached to the RAM role. Click **Complete**.

> **Note** If you want to grant, modify, or revoke the Security Token Service (STS) permissions of a RAM user, perform this step and configure the settings as required.

4. Associate the RAM user with the RAM role.
   i. In the left-side navigation pane, choose **Permissions > Policies**. On the Policies page, click **Create Policy**.
   ii.
   iii. On the Create Policy page, click the **JSON** tab. In the code editor, assign the value of the **ARN** parameter you have copied to the **Resource** parameter.



   iv. After you complete the configurations, click **OK**.
   v. In the left-side navigation pane, choose **Identities > Users**.
   vi. Find the RAM user you have created and click **Add Permissions** in the **Actions** column.
   vii. In the **Select Policy** section of the Add Permissions panel, click **Custom Policy**, select the required policies in the policy list, and then click **OK**. The permissions are granted.

5.6.2. Request a security token

To play videos based on their media IDs, you must use Security Token Service (STS) of Resource Access Management (RAM). This topic describes how to request a security token.

Prerequisites

The STS SDK that RAM provides for your programming language is installed before you request a security token. This topic uses STS SDK for Java as an example. For more information about how to install STS SDK for Java and the sample code, see STS SDK for Java. For more information about the sample code provided for STS SDKs for other programming languages, see STS SDK overview.

Context

In media workflows, each input media file is uniquely specified by a media ID. You can associate multiple output formats and resolutions with a media ID. If you want to play a media file in different formats and automatically switch between resolutions, you can play the media file based on its media ID. To play videos based on their media IDs, you must use STS of RAM.

> **Note** You must play encrypted videos based on their media IDs to ensure security.

Preparations

Before you request a security token, you must obtain the `Alibaba Cloud Resource Name (ARN)` of the role that is required by STS.

1. Log on to the RAM console. In the left-side navigation pane, choose **Identities > Roles**.
2. On the Roles page, click the name of the role whose ARN you want to obtain in the **Role Name** column. The details page of the role appears.
3. In the Basic Information section, copy the value of the **ARN** parameter. Keep the ARN confidential for later use.

1. Reference STS SDK for Java in the pom.xml file.

```
<repositories>
     <repository>
         <id>sonatype-nexus-staging</id>
         <name>Sonatype Nexus Staging</name>
         <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
         <releases>
             <enabled>true</enabled>
         </releases>
         <snapshots>
             <enabled>true</enabled>
         </snapshots>
     </repository>
 </repositories>
<dependencies>
 <dependency>
   <groupId>com.aliyun</groupId>
   <artifactId>aliyun-java-sdk-sts</artifactId>
   <version>2.1.6</version>
 </dependency>
 <dependency>
   <groupId>com.aliyun</groupId>
   <artifactId>aliyun-java-sdk-core</artifactId>
   <version>2.2.0</version>
 </dependency>
</dependencies>
```

> **Note** You can obtain the latest version of the `aliyun-java-sdk-core` package from the Maven repository.

2. Generate a security token.

```
// The main function.
public static void main(String[] args) throws Exception {
   IClientProfile profile = DefaultProfile.getProfile(
                                          "cn-hangzhou",
                                          <accessKeyId>,
                                          <accessKeySecret>);
   DefaultAcsClient client = new DefaultAcsClient(profile);
   AssumeRoleResponse response = assumeRole(client, <roleArn>);
   AssumeRoleResponse.Credentials credentials = response.getCredentials();
   System.out.println(credentials.getAccessKeyId() + "\n" +
                      credentials.getAccessKeySecret() + "\n" +
                      credentials.getSecurityToken() + "\n" +
                      credentials.getExpiration());
}
// The function that generates a temporary AccessKey pair and a security token.
private static AssumeRoleResponse assumeRole(
                                    DefaultAcsClient client,
                                    String roleArn)
                                    throws ClientException {
   final AssumeRoleRequest request = new AssumeRoleRequest();
   request.setVersion("2015-04-01");
   request.setMethod(MethodType.POST);
   request.setProtocol(ProtocolType.HTTPS);
   request.setDurationSeconds(900L);
   request.setRoleArn(roleArn);
   request.setRoleSessionName("test-token");
   return client.getAcsResponse(request);
}
```

3. Adjust the validity period of the security token.

   The security token generated in the sample code is valid for 900s. You can adjust the validity period of the security token as required. Value range: 900s to 3600s.

   During the validity period of a security token, you can use the security token without the need to generate a new one. You can use the following code to check whether a new security token needs to be generated:

```
private static boolean isTimeExpire(String expiration) {
     Date nowDate = new Date();
     Date expireDate = javax.xml.bind.DatatypeConverter.parseDateTime(expiration).getTi
me();
     if (expireDate.getTime() <= nowDate.getTime()) {
         return true;
     } else {
         return false;
     }
}
```

5.6.3. Play videos

This topic describes the methods that you can use to play videos. This topic also describes how to play videos.

Playback methods

- Play videos based on their playback URLs
  - Scenario: Videos transcoded by ApsaraVideo Media Processing (MPS) have specific output URLs. To play a transcoded video, you can directly pass its Object Storage Service (OSS) URL or Content Delivery Network (CDN) URL to ApsaraVideo Player SDK.
  - Advantage: The server generates playback URLs so that the client can directly use the URLs to play videos.
  - Disadvantage: The server needs to implement OSS authentication or CDN authentication and use domain names to concatenate playback URLs. In addition, neither automatic nor manual resolution switching can be implemented on the client.
- Play videos based on their media IDs
  - Advantage: In media workflows, each input media file is uniquely specified by a media ID. You can associate multiple output formats and resolutions with a media ID. This way, if you play a media file based on its media ID, you can play the media file in different formats and automatically switch between resolutions.
  - Playback method: Use Security Token Service (STS) of Resource Access Management (RAM) to play videos based on their media IDs. STS does not use the permissions of your Alibaba Cloud account. Instead, STS allows you to grant limited permissions with specific validity periods to RAM users or RAM roles. This prevents permission abuses and data leaks.

> **Note**
>
> - You must play encrypted videos based on their media IDs to ensure security. HTML5 players cannot play encrypted videos. We recommend that you use a Flash player to play encrypted videos on a web client.
> - You can use a media workflow to create a video-on-demand (VOD) file from a live stream. To play the VOD file, you can call the QueryMediaListByURL operation to query the media ID of the VOD file by specifying its URL in request parameters. For more information, see Transcoding SDKs.

Play videos based on their playback URLs

Implement OSS authentication or CDN authentication and use domain names to concatenate playback URLs on the server. Then, directly pass the OSS URL or CDN URL of the video to be played to ApsaraVideo Player SDK. For more information about authentication, see Playback authentication.

Play videos based on their media IDs

To play encrypted videos based on their media IDs, you must use STS of RAM. For more information about STS, see STS.

To encrypt a video, perform the following operations:

1. Construct a token issuance service to generate the MtsHlsUriToken parameter.

> **Notice** The token issuance service is used to generate the MtsHlsUriToken parameter.

2. Call the KMS decryption operation to construct a decryption service and return the DK to the player.

> **Notice** After you call a KMS operation to receive the Base64-encoded DK from KMS, you must decode the DK by using the Base64 algorithm and return the decoded DK to the player.

3. Call the QueryMediaList operation of MPS to query the OSS URL of the M3U8 file, add the MtsHlsUriToken parameter to the OSS URL, and then return the OSS URL to the player.
4. The player uses the MtsHlsUriToken parameter and the DK to require the streaming URL of the video from Alibaba Cloud CDN. Then, Alibaba Cloud CDN modifies the M3U8 file and returns the key URI and EDK to the player. Then, the player decrypts and plays the video.

STS

1. To use **STS**, you must attach required policies to a RAM role when you authorize a RAM user. For more information, see Create a RAM user and grant permissions to the RAM user.
2. After you create and authorize the RAM user, request a security token. For more information, see Request a security token. For more information about the sample code for STS SDKs for different programming languages, see STS SDK for Java, STS SDK for .NET, STS SDK for Python, STS SDK for Node.js, and STS SDK for Go.

Playback authentication

The following table describes the parameters of ApsaraVideo Player SDK.

Parameter	Description	Type	Required
vid	The media ID.	String	Yes
source	The playback URL.	String	Yes
accId	The AccessKey ID used to request a security token.	String	Yes
accSecret	The AccessKey secret used to request a security token.	String	Yes
stsToken	The security token.	String	Yes
domainRegion	The region of the media workflow, such as China (Shanghai) or China (Hangzhou).	String	Yes

Parameter	Description	Type	Required
authInfo	The authentication information. The value is a JSON string. For more information about the content of the JSON string, see the following description.	String	Yes

The value of the authInfo parameter is a JSON string that contains three arguments:

- ExpireTime: the time when the authentication expires. When you obtain the playback URL of the specified video, make sure that the authentication is valid. Otherwise, you cannot play the video. The time is displayed in UTC. Non-UTC time must be converted into UTC. Specify the time in the ISO 8601 standard in the yyyy-MM-ddTHH:mm:ssZ format, for example, 2021-03-25T23:59:59Z.
- MediaId: the media ID of the media file processed by MPS.
- Signature: the authentication signature used to filter forged requests.

  To calculate a signature, perform the following steps:

  - Set parameters in key-value pairs.

    ```
    ExpireTime="2017-03-25T23:59:59Z"
      MediaId="5aa0276ff6204ace950f75acf9e6187b"
    ```

  - Create a signature string.

    Concatenate the key-value pairs in alphabetical order with ampersands (&) to create a signature string. Note that values must be URL-encoded.

    ```
    ExpireTime=2017-03-25T23%3A59%3A59Z&MediaId=5aa0276ff6204ace950f75acf9e6187b
    ```

  - Calculate the signature.

    Calculate the HMAC_SHA1 value in binary of the preceding signature string and encode the HMAC_SHA1 value in the Base64 format.

    In UNIX-like systems, you can run the base64 command of OpenSSL to encode the calculated HMAC_SHA1 value into the Base64 format. If the authentication key is secret, you can run the following command to calculate the signature:

    ```
    $echo -n 'ExpireTime=2017-03-25T23%3A59%3A59Z&MediaId=5aa0276ff6204ace950f75acf9e6****'
    | openssl sha1 -binary -hmac 'secret' | base64
      z7mmSRuTXo4mydiWhRtbu8JKDpM=
    ```

  - Generate the signature.

    ```
    {
        "ExpireTime":"2017-03-25T23:59:59Z",
        "MediaId":"5aa0276ff6204ace950f75acf9e6****",
        "Signature":"z7mmSRuTXo4mydiWhRtbu8JKDpM="
      }
    ```

Player

MPS Player is integrated with ApsaraVideo Player. For more information, see Overview.

> **Note** You can upgrade to ApsaraVideo Player Pro as required.

5.7. Manage media libraries