

Alibaba Cloud

视频直播

API Reference

Issue: 20200604









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 due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall 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 the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. 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 bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, 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 contact Alibaba Cloud directly if you discover any errors in this document.

Document conventions

Style	Description	Example
	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.
	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.
	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.
	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 <code>cd /d C:/window</code> command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid Instance_ID</code>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>

Style	Description	Example
{ } or {a b}	This format is used for a required value, where only one item can be selected.	switch {active stand}

Contents

- Legal disclaimer..... 1**
- Document conventions.....1**
- 1 Overview of the ApsaraVideo Live API..... 1**
- 2 API authentication rules..... 5**
- 3 APIs invoke.....8**
 - 3.1 Call an API..... 8
 - 3.2 Request structure.....8
 - 3.3 Public parameters..... 9
 - 3.4 Return result..... 12
 - 3.5 Signature..... 13
- 4 Streams management..... 17**
 - 4.1 DescribeLiveStreamsOnlineList..... 17
- 5 Time-shifting live streaming..... 22**
 - 5.1 Time-shifting live streaming..... 22
- 6 Save live recordings to OSS..... 26**
 - 6.1 RealTimeRecordCommand..... 26

1 Overview of the ApsaraVideo Live API

Before using the operations, make sure that you fully understand the ApsaraVideo Live product instructions and billing methods.

The current ApsaraVideo Live OpenAPI version is 2016-11-01.

Glossary

Business and resource limits

See [Appendix](#).

Live snapshot

Snapshot setting

Operation	Description
AddLiveAppSnapshotConfig	Adds the snapshot setting.
DeleteLiveAppSnapshotConfig	Deletes the snapshot setting.

Snapshot management

Operation	Description
DescribeLiveSnapshotConfig	Queries the snapshot setting for a specified domain name.
DescribeLiveStreamSnapshotInfo	Queries snapshots.
UpdateLiveAppSnapshotConfig	Updates the snapshot setting.

Live stream pulling

Operation	Description
DeleteLivePullStreamInfoConfig	Deletes the stream pulling configuration.
AddLivePullStreamInfoConfig	Adds the stream pulling configuration.
DescribeLivePullStreamConfig	Queries the stream pulling configuration.

Stream ingest callback

Operation	Description
SetLiveStreamsNotifyUrlConfig	Adds the stream ingest callback configuration.

Operation	Description
DescribeLiveStreamsNotifyUrlConfig	Queries the stream ingest callback configuration.
DeleteLiveStreamsNotifyUrlConfig	Deletes the stream ingest callback configuration.

Live recording

Recording setting

Operation	Description
AddLiveAppRecordConfig	Adds the recording setting for a live streaming application.
DeleteLiveAppRecordConfig	Deletes the recording setting for a live streaming application.
AddLiveRecordNotifyConfig	Adds the recording callback setting.
DeleteLiveRecordNotifyConfig	Deletes the recording callback setting.
DescribeLiveRecordNotifyConfig	Queries the recording callback setting.
UpdateLiveRecordNotifyConfig	Updates the recording callback setting.

Recording management

Operation	Description
DescribeLiveRecordConfig	Queries the recording setting for a specified domain name.
DescribeLiveStreamRecordContent	Queries the recording content.
CreateLiveStreamRecordIndexFiles	Creates recording index files.
DescribeLiveStreamRecordIndexFile	Queries a single recording index file.
DescribeLiveStreamRecordIndexFiles	Queries multiple recording index files.

Live stream management

Operation	Description
ForbidLiveStream	Disables the ingest of a specified live stream .
DescribeLiveStreamsBlockList	Queries the stream ingest blacklist.
DescribeLiveStreamsControlHistory	Queries the stream control history.

Operation	Description
DescribeLiveStreamsOnlineList	Queries the list of live streams that are being ingested.
DescribeLiveStreamsPublishList	Queries the historical stream ingest records.
ResumeLiveStream	Enables the ingest of a specified live stream .
DescribeLiveStreamsFrameRateAndBitRateData	Queries the frame rate and bitrate of a live stream in real time.
DescribeLiveStreamBitRateData	Queries the historical frame rate and bitrate of a live stream within a specified duration.

Live encoding

Operation	Description
AddLiveStreamTranscode	Adds the encoding setting.
DeleteLiveStreamTranscode	Deletes the encoding setting.
DescribeLiveStreamTranscodeInfo	Queries the encoding setting.
AddCustomLiveStreamTranscode	Adds the custom encoding setting.
AddTrancodeSEI	Inserts Supplemental Enhancement Information (SEI).

Resource monitoring

Operation	Description
DescribeLiveStreamHistoryUserNum	Queries the number of historical online users of a live stream.
DescribeLiveStreamOnlineUserNum	Queries the number of real-time online users of a live stream.
DescribeLiveDomainBpsData	Queries the network bandwidth monitoring data of a live domain name.
DescribeLiveDomainTrafficData	Queries the network traffic monitoring data of a live domain name.
DescribeLiveDomainTranscodeData	Queries the encoding length data of a live domain name.
DescribeLiveDomainRecordData	Queries the recording length data of a live domain name.

Operation	Description
DescribeLiveDomainSnapshotData	Queries the number of snapshots for a live domain name.

2 API authentication rules

Introduction

Use RAM to give a subaccount access to its primary account's live video resources.

- You can activate the Live service through an Alibaba Cloud account and create a CDN domain name, then all the live functions and CDN domain names are held as resources of this account. By default, accounts have full operation permissions on their resources.
- If you want to use Alibaba Cloud Resource Access Management (RAM) service, you can grant RAM sub-users the permission to access and manage the resources under your Alibaba Cloud account.
- Read Set subaccounts to log on to the ApsaraVideo Live console by using RAM set carefully before learning how to use RAM to grant authorization and access to live service.
- If you do not need to use RAM, skip this section.

Authentication rules

When a subaccount uses live service APIs to access live resources of primary account, the live service background performs RAM access examination to make sure the resource owner grants the caller the relevant access to relevant resources.

Each different live service API determines the permissions of which resources are to be examined according to the involved resources and the meaning of API. The authentication rules of each API are listed as follows.

Action-name	Resource
DescribeLiveStreamsPublishList	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamsOnlineList	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamsBlockList	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamsControlHistory	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamOnlineUserNum	acs:cdn:*\$accountid:domain/\$domainName

Action-name	Resource
ForbidLiveStream	acs:cdn:*\$accountid:domain/\$domainName
ResumeLiveStream	acs:cdn:*\$accountid:domain/\$domainName
SetLiveStreamsNotifyUrlConfig	acs:cdn:*\$accountid:domain/\$domainName
AddLiveAppRecordConfig	acs:cdn:*\$accountid:domain/\$domainName
CreateLiveStreamRecordIndexFiles	acs:cdn:*\$accountid:domain/\$domainName
DeleteLiveAppRecordConfig	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveRecordConfig	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamRecordContent	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamRecordIndexFile	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamRecordIndexFiles	acs:cdn:*\$accountid:domain/\$domainName
AddLiveStreamTranscode	acs:cdn:*\$accountid:domain/\$domainName
DeleteLiveStreamTranscode	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveStreamTranscodeInfo	acs:cdn:*\$accountid:domain/\$domainName
AddLiveAppSnapshotConfig	acs:cdn:*\$accountid:domain/\$domainName
DeleteLiveAppSnapshotConfig	acs:cdn:*\$accountid:domain/\$domainName
UpdateLiveAppSnapshotConfig	acs:cdn:*\$accountid:domain/\$domainName
DescribeLiveSnapshotConfig	acs:cdn:*\$accountid:domain/\$domainName

Action-name	Resource
DescribeLiveStreamSnapshotInfo	acs:cdn:*:\$accountid:domain/\$domainName
AddLiveSnapshotDetectPornConfig	acs:cdn:*:\$accountid:domain/\$domainName
AddLiveDetectNotifyConfig	acs:cdn:*:\$accountid:domain/\$domainName
DescribeLiveSnapshotDetectPornConfig	acs:cdn:*:\$accountid:domain/\$domainName
DescribeLiveDetectNotifyConfig.md	acs:cdn:*:\$accountid:domain/\$domainName
UpdateLiveSnapshotDetectPornConfig	acs:cdn:*:\$accountid:domain/\$domainName
UpdateLiveDetectNotifyConfig	acs:cdn:*:\$accountid:domain/\$domainName
DeleteLiveSnapshotDetectPornConfig	acs:cdn:*:\$accountid:domain/\$domainName
DeleteLiveDetectNotifyConfig	acs:cdn:*:\$accountid:domain/\$domainName
DescribeLiveStreamsFrameRateAndBitRateData	acs:cdn:*:\$accountid:domain/\$domainName

3 APIs invoke

3.1 Call an API

You can call an ApsaraVideo Live API by sending an HTTP GET request to the ApsaraVideo Live API server and adding relevant request parameters to the request according to the interface instructions. The system returns the result based on the processing of the request.

The instructions for the four aspects are as follows:

- [Request structure](#)
- [Public parameter](#)
- [Returned result](#)
- [Signature](#)

3.2 Request structure

Service address

The API service access address is as follows:

Region	Service address
China East 2	live.aliyuncs.com
China North 2	
Singapore	

**Note:**

live.aliyuncs.com is ApsaraVideo Live endpoint.

Communication protocol

We recommend that you send requests using the HTTPS channel for enhanced security.

Character encoding

Both requests and returned results are encoded using the UTF-8 character set.

3.3 Public parameters

Public parameters refers to the parameters that every interface call uses, including public request parameters and public response parameters.

Public request parameters

Public request parameters refer to the request parameters that every interface uses.

Parameters	Type	Required	Description
Version	String	Yes	API version. <ul style="list-style-type: none"> The format is YYYY-MM-DD. The current version is 2016-11-01.
AccessKeyId	String	Yes	The Key ID provided by Alibaba Cloud for you to access services.
Signature	String	Yes	The request signature. See Signature for the signature calculation method.
SignatureMethod	String	Yes	The mode of the signature. HMAC-SHA1 is supported currently.

Parameters	Type	Required	Description
Timestamp	String	Yes	<p>The timestamp to request.</p> <ul style="list-style-type: none"> The date format follows the ISO8601 standard and uses UTC time . Format: YYYY-MM-DDThh:mm:ssZ. Example: 2014-11-11T12:00:00Z (equivalent to 20:00:00 on November 11, 2014, Beijing time).
SignatureVersion	String	Yes	Signature algorithm version. The current version is 1.0.
SignatureNonce	String	Yes	The unique random number that is used to prevent network replay attacks. You must use different random numbers for different requests.

Parameters	Type	Required	Description
ResourceOwnerAccount	String	No	The resource owner account for this API request, that is the login user name. For more information about using this parameter, see RAM resource authorization .(You can only use this parameter in the Action of the RAM in which the live resources can be authorized, or the request is rejected.)
Format	String	No	Type of value returned, JSON and XML supported. Default value: XML.

Examples

```
https://live.aliyuncs.com/
? Format=xml
&Version=2016-11-01
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgl%3D
&SignatureMethod=HMAC-SHA1
&SignatureNonce=15215528852396
&SignatureVersion=1.0
&AccessKeyId=key-test
&Timestamp=2012-06-01T12:00:00Z
...
```

Public response parameters

Each time you make a call to an API, the system returns a unique identification code (RequestId), regardless whether the request is successful.

Examples

- XMLexample

```
<? xml version="1.0" encoding="UTF-8"? >
<! --Result Root Node-->
<Interface Name+Response>
<! --Return Request Tag-->
<RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
```

```
<!--Response Result Data-->
</Interface Name+Response>
```

- JSON example

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
  /* Returned data*/
}
```

3.4 Return result

The data returned after the API service call adopts a uniform format:

If the returned HTTP status code is 2xx, it indicates the call is a success. If the returned HTTP status code is 4xx or 5xx, then it is a failed call. The two primary formats for the data returned after a successful call are XML and JSON. The external system can customize the returned data format by passing in a parameter in the request, and XML is adopted by default.



Note:

In the returned examples in this document, we adjusted the format of returned result to make it easier for you to view the content. The actual returned result does not go through the line breaks, indentation, or other layouts.

Successful results

JSON example

```
{
  "RequestId": "16A96B9A-F203-4EC5-8E43-CB92E68F4CD8"
}
```

Error results

When an error occurs in an interface call, no result data is returned. You can see the error code corresponding to each interface and the Error Code Table as follows to locate the cause of the error. When an error occurs in a call, an HTTP status code of 4xx or 5xx is returned for the HTTP request. The returned message body contains the specific error code and error message. The message body also contains a globally unique RequestId and the requested HostId. If you fail to locate the cause of the error, contact our customer service and provide the HostId and RequestId for us to solve the problem as quickly as possible.

JSON example

```
{
```

```
"Code": "InternalError",
"HostId": "live.aliyuncs.com",
"Message": "The request processing has failed due to some unknown error.",
"RequestId": "6EBD1AC4-C34D-4AE1-963E-B688A228BE31"
}
```

Public error code table

For more information about error code, see [API error center](#).

3.5 Signature

Alibaba Cloud performs identity authentication for every access request. Therefore, you must contain the signature information in the request no matter whether you submit a request through the HTTP or HTTPS protocol. The requester identity is verified using symmetric encryption of the `Access Key ID` and `Access Key Secret`. The `Access Key ID` and `Access Key Secret` are officially issued to visitors by Alibaba Cloud (you can apply for and manage them on the Alibaba Cloud official website). In specific, the `Access Key ID` indicates the identity of the visitor. The `Access Key Secret` is the secret key used to encrypt the signature string and to verify the signature string on the server. It must be kept strictly confidential and only be known to Alibaba Cloud and the user.



Note:

Alibaba Cloud offers SDKs and third-party SDKs in different languages, which free you of the trouble of coding the signature algorithm. For more information about Alibaba SDK, see [Alibaba Cloud SDK](#).

Signature operation

During access, use the following methods to sign a request.

1. Use request parameters to construct the canonicalized query string.
 - a. Sort parameters.

All the request parameters (including the public request parameters and user-defined parameters with given request interfaces, but excluding the `Signature` parameter mentioned in the public request parameters) are sorted alphabetically by the parameter name.



Note:

Note:When a request is submitted using the GET method, these parameters are the parameter section of the request URI (that is, the section in the URI following? and connected by &).

b. Encode parameters.

The name and value of each request parameter are encoded. The names and values must adopt UTF-8 characters for URL encoding. The URL encoding rules are as follows:

- The characters A-Z, a-z, 0-9, and -, _, ., ~ are not encoded.
- Other characters are encoded into the %XY format, with XY representing the characters' ASCII code in hexadecimal notation. For example, the English double quotation marks (") are encoded as %22.
- Extended UTF-8 characters are encoded into the %XY%ZA... format.
- The English space () is encoded as %20, rather than the plus sign (+).

This encoding method and the commonly-used application/x-www-form-urlencoded MIME type (such as java.net.URLEncoder in Java library) are similar, but have differences. If this encoding method is used, use the method of standard library to encode, and then replace the plus signs (+) in the encoded strings with %20, the asterisks (*) with %2A, and change %7E back to the tilde (~) to get the encoded string described in the previous rules. This algorithm can be achieved by using the following method:

```
private static final String ENCODING = "UTF-8";

private static String percentEncode(String value) throws UnsupportedEncodingException {
    return value != null ? URLEncoder.encode(value, ENCODING).replace("+", "%20").
        replace("*", "%2A").replace("%7E", "~") : null;
}
```

- c.** Connect the encoded parameter names and values with the English equals sign (=).
- d.** Then, sort the parameter name and value pairs connected by equal signs in alphabetical order and connect them with the & symbol to produce the Canonicalized Query String.

2. Construct the string for signature calculation using the canonicalized query string in the previous step according to the following rules.

```
StringToSign=
HTTPMethod + "&" +
percentEncode( "/" ) + " &" +
```

```
percentEncode(CanonicalizedQueryString)
```

Wherein,

- HTTPMethod is the HTTP method used for request submission, for example, GET.
 - percentEncode("/") is the encoded value (%2F) of the character /, which is obtained according to the URL encoding rules described in 1.ii.
 - percentEncode(CanonicalizedQueryString) is the Canonicalized Query String (constructed in Step 1) that is encoded according to the URL encoding rules described in 1.ii.
3. Use the previous signature string to calculate the signature's HMAC value based on RFC2104 definitions.

**Note:**

Note: The key used for signature calculation is the Access Key Secret held by the user plus the & character (ASCII:38), and the SHA1 hashing algorithm is used.

4. Encode the previous HMAC value into a string based on Base64 encoding rules to obtain the signature value (Signature).
5. Add the obtained signature value to the request parameters as the Signature parameter to sign the request.

**Note:**

Note: URL encoding is required to be performed for the obtained signature value based on the [RFC3986](#) rule, like in the case of other parameters, before the signature value is submitted to the live server as the final request parameter value.

Examples

Take **DescribeLiveSnapshotConfig** as an example, the request URL before signing is as follows:

```
http://live.aliyuncs.com/?Format=XML&SignatureMethod=HMAC-SHA1&Action=DescribeLiveSnapshotConfig&AccessKeyId=testid&RegionId=cn-shanghai&ServiceCode=live&DomainName=test.com&AppName=test&SignatureNonce=c2fe8fbb-2977-4414-8d39-348d02419c1c&Version=2016-11-01&SignatureVersion=1.0&Timestamp=2017-06-14T09:51:14Z
```

The StringToSign is:

```
GET%2F&AccessKeyId%3Dtestid&Action%3DDescribeLiveSnapshotConfig&AppName%3Dtest&DomainName%3Dtest.com&Format%3DXML&RegionId%3Dcn-shanghai&ServiceCode%3Dlive&SignatureMethod%3DHMAC-SHA1&SignatureNonce%3Dc2fe8fbb
```

```
-2977-4414-8d39-348d02419c1c&SignatureVersion%3D1.0&Timestamp%3D2017-06-14T09%253A51%253A14Z&Version%3D2016-11-01
```

Assume the Access Key Id is testid, the Access Key Secret is testsecret, and the Key used for HMAC calculation is testsecret, the calculated signature value is:

```
315a3myPjp8FXWT4rvxX5pKb/aw=
```

The signed request URL is (note the added Signature parameter):

```
http://live.aliyuncs.com/?Format=XML&SignatureMethod=HMAC-SHA1&Signature=315a3myPjp8FXWT4rvxX5pKb%2Faw%3D&Timestamp=2017-06-14T09%3A51%3A14Z&Action=DescribeLiveSnapshotConfig&AccessKeyId=testid&RegionId=cn-shanghai&ServiceCode=live&DomainName=test.com&AppName=test&SignatureNonce=c2fe8fbb-2977-4414-8d39-348d02419c1c&Version=2016-11-01&SignatureVersion=1.0
```


4 Streams management

4.1 DescribeLiveStreamsOnlineList

View the information of all the streams being pushed under a specific domain name (or an app under a specified domain name).




Note:

The access frequency of DescribeLiveStreamsOnlineList is limited to 1,000 times per minute. We recommend that you do not call it frequently, so as not to cause the business to be unavailable.

Request parameters

Parameters	Type	Required	Example values	Description
Action	String	Yes	DescribeLiveStreamsOnlineList	This interface. Value: DescribeLiveStreamsOnlineList
DomainName	String	Yes	play. yourdomain. com	Your Live domain name.
AppName	String	No	testApp	Name of the app. <div data-bbox="1209 1429 1278 1498" data-label="Image"> </div> <div data-bbox="1294 1458 1380 1491" data-label="Section-Header"> <p>Note:</p> </div> <div data-bbox="1206 1498 1426 1704" data-label="Text"> <p>AppName does not support wildcard query (*) and fuzzy query.</p> </div>

Parameters	Type	Required	Example values	Description
StreamName	String	No	livestream	Stream name.  Note: StreamName does not support wildcard (*) queries, but supports fuzzy queries.
StreamType	String	No	all	Stream type. Valid values: <ul style="list-style-type: none"> • all • raw • trans Correspondingly check all the streams, raw streams and transcoding streams. Default: all means returning all the stream information.
QueryType	String	No	fuzzy	Specifies whether the stream name is fuzzy matching. Value: <ul style="list-style-type: none"> • fuzzy: fuzzy matching • strict: precise matching

Parameters	Type	Required	Example values	Description
StartTime	String	No	2016-06-29T19:00:00Z	Start time. <ul style="list-style-type: none"> • UTC format • Example: 2016-06-29T19:00:00Z
EndTime	String	No	2016-06-30T19:00:00Z	EndTime. <ul style="list-style-type: none"> • UTC format • Example: 2016-06-30T19:00:00Z • The interval between EndTime and StartTime cannot exceed 30 days.
PageNum	Integer	No	1	The page number. Default value: 1.
PageSize	Integer	No	1,500	Size of each page. Maximum value: 3000. Value: Any integer between 1 and 3000. Default value: 2000

Response parameters

Parameters	Type	Example values	Description
OnlineInfo			Information of the stream being pushed

Parameters	Type	Example values	Description
↳ DomainName	String	play.yourdomain.com	The Live domain name, which the stream belongs to.
↳ AppName	String	AppName	Name of the app, which the stream belongs to.
↳ StreamName	String	StreamName	Name of the stream.
↳ PublishTime	String	2015-12-02T06:58:04Z	The time when stream ingest starts in UTC.
↳ PublishUrl	String	rtmp://play.aliyunlive.com/AppName/StreamName	The full ingest URL.
↳ PublishDomain	String	push.aliyunlive.com	The ingest domain. If using the live center ingest, you can enter the streaming domain directly.
RequestId	String	40A4F36D-A7CC-473A-88E7-154F92242566	The ID of the job request.
PageNum	Integer	1	The page number.
PageSize	Integer	10	The page size.
TotalNum	Integer	10	The total number that conforms with the conditions.
TotalPage	Integer	100	The total number of pages.

Example

Request example

```
https://live.aliyuncs.com/?Action=DescribeLiveStreamsOnlineList&DomainName=test101.aliyunlive.com&PageSize=10&PageNum=2&<Public Request Parameter>
```



Note:

For more information, see [Public Request Parameter](#).

Normal response example

JSON format

```
{
  "OnlineInfo":{
    "LiveStreamOnlineInfo":[{
      "AppName":"xchen",
      "DomainName":"test101.cdnpe.com",
      "PublishTime":"2015-12-02T06:58:04Z",
      "PublishUrl":"rtmp://test101.cdnpe.com/xchen",
      "StreamName":"testxchen"
    }]
  },
  "PageNum":2,
  "PageSize":10,
  "RequestId":"0D70427D-91E4-4349-AAD3-5511A5BB823B",
  "TotalNum":11,
  "TotalPage":2
}
```

Exception response example

JSON format

```
{
  "Code":"InternalError",
  "HostId":"live.aliyuncs.com",
  "Message":"The request processing has failed due to some unknown error.",
  "RequestId":"6EBD1AC4-C34D-4AE1-963E-B688A228BE31"
}
```

Error code

For more information about error code of this product, see [Error code](#).

5 Time-shifting live streaming

5.1 Time-shifting live streaming

Principle of time-shifting live streaming

Time-shifting live streaming is based on conventional HTTP Live Streaming (HLS). In live streaming, an ingest stream is segmented into .ts files, which are distributed to playback users through the HLS protocol. The .m3u8 playlist files requested by users contain constantly refreshing .ts file addresses. In conventional HLS, the .ts file addresses and corresponding .ts files are not persistently stored, making it impossible to trace the live streaming video content prior to the current time. In contrast, in HLS with the time shifting feature enabled, the .ts file addresses and corresponding .ts files are persistently stored in the database and OSS bucket independently for up to 15 days. This makes it possible to trace the video content back from the start time of live streaming to the current time.

Time shifting setting

The time shifting feature is now available in open beta. You can open a ticket to try this feature.

Item	Description
Enterprise name	None
Streaming domain	All live streams under the specified streaming domain support time shifting.
Retention period of time-shifting content	Time-shifting content is retained for seven days by default.

Request parameters for time shifting

To implement time-shifting live streaming, you need to add time-shifting parameters to a conventional HLS URL such as `http://domain/app/stream.m3u8`.

Required parameter

`lhs_start=1`

**Note:**

You must specify this parameter to call a time-shifting operation through CDN.

Time-related parameter

The format of the key is `lhs_{type}_{format}_{unit}_{zone}`.



Note:

The parameter name consists of four variables and a fixed prefix `lhs_`. The variables are separated with underscores (`_`). The variables are described as follows:

- **type:** the type of the time. Valid values:
 - `start`: the start time of live streaming.
 - `end`: the end time of live streaming.
 - `vodend`: the end time of video-on-demand (VOD). If you set this variable to `vodend`, the video is played back in VOD mode. In this case, all `.ts` files within a specified duration are returned at a time, including the endlist tags.
 - `offset`: the forward offset time.
- **format:** the format of the time. Valid values:
 - `unix`: the UNIX timestamp.
 - `human`: the human-readable time such as 20170809230130.
- **unit:** the unit of the time. Valid values:
 - `s`: second
 - `ms`: millisecond
 - `us`: microsecond
 - `ns`: nanosecond
- **zone:** the time zone.
 - Valid values: any integer from 0 to 9. The value indicates the number of hours to add to UTC to get the standard time in the specified time zone. A value of 0 indicates the UTC time. A value of 8 indicates the China Standard Time.

Examples

- `lhs_start_unix_s_0=1502280113`
- `lhs_offset_unix_ms_8=1502308959375`
- `lhs_end_human_s_8=20170809200010`

Sample requests

```
http://xxx.com/live/channel1.m3u8?lhs_start=1&lhs_start_human_s_8=20171024160220&lhs_end_human_s_8=20171024160420"
```

Timeline query

You can query the periods of time during which time-shifting data was involved within a historical duration along the timeline of a specified stream.

- URL: `http://{$domain}/openapi/timeline/query`
- Method: GET
- Parameters:
 - `app` (required): the name of the live streaming application.
 - `stream` (required): the name of the live stream.
 - `format` (required): the format of the time-shifting data to be queried, which can be `.ts` or `.flv`.
 - `lhs_start_unix_s_0` (required): the start timestamp, measured in seconds since 00:00:00 Thursday, 1 January 1970.
 - `lhs_end_unix_s_0` (required): the end timestamp, measured in seconds since 00:00:00 Thursday, 1 January 1970.



Note:

If you have enabled CDN authentication, you must add the authentication parameters accordingly.

- Response (JSON format):
 - `current`: the current system time, with which the time of your player is aligned.
 - `timeline`: an array. Each element of the array is called a map, which indicates a valid time-shifting period. Each map includes the following fields:
 - `timeline[i].start`: the start timestamp of a valid `.ts` file, measured in seconds since 00:00:00 Thursday, 1 January 1970.
 - `timeline[i].end`: the end timestamp of a valid `.ts` file, measured in seconds since 00:00:00 Thursday, 1 January 1970.

- Example

```
http://$host/openapi/timeline/query? app=bbb&domain=aaaa&format=ts&
lhs_end_unix_s_0=1513334270&lhs_start_unix_s_0=1513334255&stream=cc
```

Response

```
{"retCode":0,"description":"success","content":{"current":1514269063,"timeline":[{"start
":1514269054,"end":1514269058},{"start":1514269058,"end":1514269062}]}}
```

6 Save live recordings to OSS

6.1 RealTimeRecordCommand

You can call this operation to complete manual recording on demand. You can start and stop recording at any time.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to RealTimeRecordCommand .
Command	String	Yes	The operation. Valid values: start and stop.
DomainName	String	Yes	The CDN domain name.
AppName	String	Yes	The name of the live streaming application.
StreamName	String	Yes	The name of the live stream.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.

Examples

Sample requests

```
https://live.aliyuncs.com?Action=RealTimeRecordCommand&DomainName=test101.cdnpe.com&AppName=abc&StreamName=s1&Command=start<Common request parameters>
```

Sample responses

JSON format

```
{
  "RequestId": "16A96B9A-F203-4EC5-8E43-CB92E68F4CD8"
}
```

Error codes

Error code	Error message	HTTP status code	Description
InvalidDomain.NotFound	The domain provided does not exist in our records.	404	The error message returned when the domain name does not exist or does not belong to the current user.
MissingDomainName	DomainName is mandatory for this action.	400	The error message returned when the value of DomainName is missing.
MissingAppName	AppName is mandatory for this action.	400	The error message returned when the value of AppName is missing.
MissingStreamName	StreamName is mandatory for this action.	400	The error message returned when the value of StreamName is missing.
MissingCommand	Command is mandatory for this Command.	400	The error message returned when the value of Command is missing.
InvalidStream.NotFound	Stream does not exist.	404	The error message returned when the specified stream does not exist.
InvalidConfig.NotFound	Config does not exist.	404	The error message returned when the specified setting does not exist.

Error code	Error message	HTTP status code	Description
TaskAlreadyStarted	Task has already started.	409	The error message returned when the recording task has already started.
InternalServerError	The request processing has failed due to some unknown error, exception or failure.	500	The error message returned when an internal error occurred.

For more information, see [common error codes of ApsaraVideo Live](#).