



Microsoft Azure

Azure Media Services Connector

TABLE OF CONTENTS

TABLE OF CONTENTS	1
1 OVERVIEW	1
1.1 FEATURES	2
1.2 GETTING STARTED	2
1.3 SERVICE OVERVIEW	2
2 API REFERENCE	3
2.1 HEADERS SUPPORTED BY ALL APIS	3
2.1.1 ACCEPT REQUEST HEADER	3
2.1.2 CONTENT-TYPE REQUEST/RESPONSE HEADER	3
2.2 HTTP STATUS CODES	3
2.3 SERVICE APIS	4
2.3.1 INTRODUCTION	4
2.3.2 VIDEOS	4
2.3.3 THUMBNAILS	10
2.3.4 PLAYBACK	11
2.3.5 FORMATS	12
2.4 UPLOADING ENCRYPTED CONTENT	13
2.5 UPLOAD VIDEO TO AZURE BLOB	13
3 END-2-END WORKFLOW	13
3.1 SCENARIOS	13
3.1.1 (UNENCRYPTED) CONTENT CREATION AND PLAYBACK SCENARIO	13
3.1.2 ENCRYPTED CONTENT CREATION AND PLAYBACK SCENARIO	14
4 REPORTING ISSUES	14

1 OVERVIEW

In today's fast paced environment, enterprises are looking for feature enhancements that offer more flexibility and make the lives of their employees easier and more productive. Today, we are advancing our offering to deliver features that do just that – help customers in their business transformation journey. The goal is to enable enterprise developers, ISVs and employees to quickly and easily build video enabled applications that require minimal coding expertise, and that can scale cost effectively.



With Azure Media Services Connector you get a simple and powerful interface to access Azure Media Services. With this you will be able to create end-to-end media workflows which abstracts the complexities of the video pipeline and allows you to focus on building value-added solutions.

1.1 FEATURES

- **Simple APIs** – Simple REST APIs provide access to scalable video that can be played back on any screen. Our REST APIs abstract the complexities of the video pipeline and lets ISVs and Enterprise professional developers to focus on building value-added solutions.
- **Reach** – Extend your audience reach by streaming pre-recorded audio and video content to virtually any player and device – simultaneously, over any protocol, and from a single set of source files.
- **High-quality Playback with Adaptive Streaming** – Stream adaptively to deliver the best possible playback quality on Android, iOS, Flash, and other clients anywhere in the world, regardless of bandwidth conditions. Increase viewer satisfaction, save on transcoding costs, and simplify the management of video files.
- **Security** – The power of video has not gone unnoticed by enterprises. Every day, more and more companies are integrating video into their daily processes. However, while embracing video can successfully increase efficiency and communication in the workplace, it goes without saying that the security of the content is of immense importance in an enterprise setting. With Azure Media Services Connector, enterprises can be assured that their content is always secure. We use state-of-the-art AES-128 bit encryption coupled with secure token exchange before a video can be played.
- **Beyond Video** – As we continue to add new features and capabilities to our platform, our customers will benefit from being able to build more advanced video solutions which solve real business problems. We will continue to bring those to you through the Azure Media Services Connector.

1.2 GETTING STARTED

REST APIs supported by Azure Media Services Connector offer developers a means to manage a highly scalable video content store. This connector is designed to support the integration of video content into a client Content Management System (CMS). Services are accessible via REST APIs directly over the internet from any application that can make HTTPS requests and interpret HTTPS responses. Azure Media Services Connector supports only HTTPS endpoints; HTTP is not supported.

Azure Media Services Connector enables media encoding and streaming capabilities through an Azure Media Services service instance (account). When creating a new connector instance you will need to provide a name for the Media Services instance to be created in the 'Package Settings' section. The name must meet the following criteria:

- Globally unique (no other Media Service instance can have the same name across Microsoft Azure);
- Lowercase letters and numbers only;
- Between 3 and 24 characters of length.

The uniqueness validation only occurs once the creation is confirmed and may take few minutes to complete. Interactive name validation will be provided at a later time.

Media Services Connector data model is exposed through a REST API.

1.3 SERVICE OVERVIEW

Within a Media Services account, a client application can create and manage Video entities. After a Video entity is created, client application will upload the video content to an Azure BLOB container managed by Media Services. The client then indicates to Media Services that uploading is completed. Once that occurs, Media Services encodes the input video into a H.264 format. It also creates thumbnails for each stored video. Once the Status endpoint of the Media Connector indicates success **in the body of the response**, the video is ready for streaming.

Media Services supports playback streaming formats as shown:

Streaming Formats Supported by Media Services

Apple HTTP Live Streaming



Media Services supports encrypted delivery of video content to compatible devices. Encrypted delivery can be configured at video creation time and is implemented using an AES encrypted stream delivered over HTTP.

2 API REFERENCE

2.1 HEADERS SUPPORTED BY ALL APIS

Headers in this section are supported by all APIs.

2.1.1 ACCEPT REQUEST HEADER

The Accept request header controls the format of response data produced by Media Services. Clients may specify formats as shown. Media Services Connector defaults to 'application/json' responses if not specified.

Request data formats supported by Media Services Connector

application/json

2.1.2 CONTENT-TYPE REQUEST/RESPONSE HEADER

The Content-Type header specifies the media type of request or response content.

Supported Content Types

application/json

2.2 HTTP STATUS CODES

Each API call to Media Services Connector can result in one of the HTTP status codes listed below. Transient status codes indicate conditions that could resolve themselves with the passage of time, and could be retried. We recommend using appropriate retry policies to handle such transient conditions.

Status	Condition	Method	Transient
200	OK	GET	n/a
201	Created	POST	n/a
202	Accepted	POST	n/a
204	No content	POST, PUT, DELETE	n/a
307	Temporary Redirect	GET	n/a
304	Not Modified	Conditional GET	n/a
400	Bad request	All	No
401	Unauthorized; authentication not provided	All	No
403	Forbidden; authenticated but not authorized	All	No
404	Not found	All	No
405	(Method) Not allowed	All	No
406	Not acceptable - Accept header doesn't match a response type supported by the server	GET	No
408	Timeout	all	Yes
409	Resource conflict - state of resource doesn't allow modification	PUT, POST, DELETE	Yes
412	Precondition failed	PUT	No
413	Request entity too large -- size of request exceeded server limit	POST, PUT	No



414	Request URI too long	POST, PUT	No
415	Unsupported type -- representation not supported	POST, PUT	No
500	Server error (internal) This should be used only in situations that are not covered by above options.	All	Yes
501	Not implemented	All	No
503	Service Unavailable	All	Yes

2.3 SERVICE APIS

2.3.1 INTRODUCTION

Azure Media Services Connector exposes the following resources via the REST API:

- **Video** – The Videos entity represents the set of videos available within a given account
- **Thumbnail** – A Thumbnail entity represents an image derived from a frame within a Video.
- **Formats** –The Formats endpoint is used to retrieve a list of supported input video formats.

Each resource may support GET, PUT, POST, or delete methods, based on the scenarios supported. For purposes of this document, examples assume that the path portion of the service root URL is /api.

2.3.2 VIDEOS

A Video is the primary entity in our service representing a video. Its entity path is Videos.

2.3.2.1 PROPERTIES

Property	Type	Description	Required?
ContentProtectionKeyId	String	If ContentProtectionMode="AES128", this property contains the ID of the key used to secure the video.	
ContentProtectionMode	String	Controls the protection mode used when streaming the video for playback. Expected values: "None" or "AES128"	
Duration	int	Duration of video in seconds, if available.	
ID	String	Primary key of entity	
InputStorageEncryptionMode	String	Encryption mode for input video. Expected values: empty or "AES256".	
InputStorageEncryptionSymmetricKey	String	The symmetric key to be used by the client to encrypt video content before uploading, base 64 encoded. This value is present only on a Video entity returned by a POST operation.	
InputStorageEncryptionIv	String	The initialization vector to be used by the client to encrypt video content before uploading, base 64 encoded. This value is present only on a Video entity returned by a POST operation.	
OriginalFileName	String	The name of the video file which you are uploading. Once the file is uploaded, we will rename it to <guid>.<file-extension> in the response back.	Y
UploadUri	String	This is the URI where video content should be uploaded to (an Azure Storage location managed by Media Services). Once you have uploaded a video and indicated that to us, it is no longer modifiable and this URL is empty. Please refer code sample section for details.	



2.3.2.2 SUPPORTED OPERATIONS

2.3.2.2.1 RETRIEVE A VIDEO

Method	Request URI	HTTP Version
GET	/api/Videos/{id}	HTTP/1.1

Sample Request

Request Headers

```
GET https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/fc68482e-c4e2-49cf-b39f-56264a4e2ddf HTTP/1.1
```

Request Body

None

If successful, a **200 OK** status code is returned along with a representation of the created entity in the response body.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 307
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl: https://
testazuremediaservicesconnector.azurewebsites.net/api/Videos/fc68482e-c4e2-49cf-b39f-56264a4e2ddf
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
Date: Mon, 23 Mar 2015 19:31:39 GMT

{"Id":"fc68482e-c4e2-49cf-b39f-56264a4e2ddf","UploadUri":null,"ContentProtectionMode":"None","ContentProtectionKeyId":null,"Duration":13,"OriginalFileName":"1141fb2a-af46-43c9-aa13-f0b49c7c13ae.mp4","InputStorageEncryptionMode":null,"InputStorageEncryptionSymmetricKey":null,"InputStorageEncryptionIv":null}
```

If the video ID is incorrect, then you get **404 – Not Found**

```
HTTP/1.1 404 Not Found
```

2.3.2.2.2 CREATE A VIDEO

Method	Request URI	HTTP Version
POST	/api/Videos	HTTP/1.1

Sample Request/Response

Request Headers

```
POST https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/ HTTP/1.1
Content-Type: application/json
```

Request Body

```
{
  "OriginalFileName": "yourfilename.mp4"
}
```

If successful, a **200 OK** status code is returned along with a representation of the created entity in the response body.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
```



```

Pragma: no-cache
Content-Length: 611
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl: https://testazuremediaservicesconnector.azurewebsites.net/api/Videos
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 19:35:07 GMT

{
  "Id": "ee0dfd10-4a38-469f-98dc-2c8e34de0be8",
  "UploadUri": "https://cvprbl203m01r12.blob.core.windows.net:443/asset-0914435d-1500-80c3-9b0e-
fle4d193b00f/a6dad4be-6a34-423d-85de-b43491133140.mp4?sv=2012-02-12&sr=c&si=5e9ce4f6-9075-40d1-929a-
831ebdc1edd9&sig=61s0Ebc%2FWsz9zaJYh%2FOnsj3%2FG7Mbzo%2FuBDARATEKQcs%3D&st=2015-03-
23T19%3A30%3A07Z&se=2015-03-24T08%3A00%3A07Z",
  "ContentProtectionMode": "None",
  "ContentProtectionKeyId": null,
  "Duration": 0,
  "OriginalFileName": "a6dad4be-6a34-423d-85de-b43491133140.mp4",
  "InputStorageEncryptionMode": null,
  "InputStorageEncryptionSymmetricKey": null,
  "InputStorageEncryptionIv": null
}

```

If you don't provide an OriginalFileName, you will get a **400 – Unable to deserialize JSON into Video object**

```

HTTP/1.1 400 Unable to deserialize JSON into Video object...
HTTP/1.1 400 Unable to deserialize JSON into Video object. You must minimally specify OriginalFileName
field. FileName doesn't matter but you must provide CORRECT extension.
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 478
Content-Type: text/html; charset=Windows-1252
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl: https://testazuremediaservicesconnector.azurewebsites.net/api/Videos
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 19:40:05 GMT

<h1>Error</h1> <table> <tbody> <tr> <td><strong>Request</strong> </td>
<td>https://azuremediaservicesconnectorc0691ddba6f2424ea4aa3e4644091464.azurewebsites.net/api/Videos</t
d> </tr> <tr> <td><strong>Error</strong></td> <td>BadRequest</td> </tr> <tr>
<td><strong>Message</strong></td> <td>Unable to deserialize JSON into Video object. You must minimally
specify OriginalFileName field. FileName doesn't matter but you must provide CORRECT extension.</td>
</tr> </tbody> </table>

```

If ContentProtectionMode is anything other than 'AES128' or 'empty' or if InputStorageEncryptionMode is anything other than AES256 or 'empty', you will get a **400 Bad Request**

```

HTTP/1.1 400 Bad Request
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 173
Content-Type: application/json; charset=utf-8
Server: Microsoft-IIS/8.0

```



```

Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl: https://testazuremediaservicesconnector.azurewebsites.net/api/Videos
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 19:42:14 GMT

{
  "status": 0,
  "source": "https://testazuremediaservicesconnector.azurewebsites.net/api/Videos",
  "message": "The request is invalid."
}

```

2.3.2.2.3 DELETE A VIDEO

Method	Request URI	HTTP Version
DELETE	/api/Videos/{id}	HTTP/1.1

Sample Request/Response

Request Headers

```
DELETE https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/4f13e32f-7eab-4c24-948a-
f817647dbe2b HTTP/1.1
```

Request Body

None

If the video is successfully deleted or if you do not upload the video after calling create video, you get a **204 – No Content**

```

HTTP/1.1 204 No Content
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 0
Expires: -1
Vary: Content-Type
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-Content-Type-Options: nosniff
x-ms-svc-ver: 0.0.1503.1901
x-ms-request-elapsed-ms: 39
client-request-id: ceeba89a-8d64-47fe-bbb2-9e0087139fdd
request-id: 20bd48f5-88f4-41ee-8ef0-41dba8bd626f
ocp-cloudvideo-instance-id: VideoManagementService IN 1
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl: https://
testazuremediaservicesconnector.azurewebsites.net/api/Videos/ee0dfd10-4a38-469f-98dc-2c8e34de0be8
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 19:44:22 GMT

```

If you specify a wrong video id, you will get a **404 – Not Found**

```

HTTP/1.1 404 Not Found
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 0
Expires: -1
Vary: Content-Type

```



```

Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-Content-Type-Options: nosniff
x-ms-svc-ver: 0.0.1503.1901
x-ms-request-elapsed-ms: 27
client-request-id: 4a77af4e-4a0c-45fb-8056-f3ba903acb0
request-id: bc9f67e4-dbc1-430a-aae5-f3dd2a7dcf63
ocp-cloudvideo-instance-id: VideoManagementService IN 0
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/ee0dfd10-4a38-469f-asddsa98dc-
2c8e34de0be8
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 19:45:37 GMT

```

2.3.2.2.4 MARK COMPLETE

Clients signal that they have completed uploading to BLOB storage by calling the MarkComplete action on a video

Method	Request URI	HTTP Version
POST	POST /api/Videos/{id}/MarkComplete	HTTP/1.1

Sample Request/Response

Request Headers

```

POST https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/ee0dfd10-4a38-469f-
asddsa98dc-2c8e34de0be8/MarkComplete HTTP/1.1

```

Request Body

None

If successful, a **200 OK** status code is returned.

```

HTTP/1.1 200 OK
Cache-Control: no-cache

```

If you submit a wrong video id for Mark Complete, you will get a **404 – Not Found**

```

HTTP/1.1 404 Not Found
Cache-Control: no-cache

```

2.3.2.2.5 AUTHORIZATION TOKEN

Clients retrieve a key token for use in the AES128 content protection mode playback interaction by calling the AuthorizationToken endpoint

Method	Request URI	HTTP Version
GET	GET /api/Videos/{id}/AuthorizationToken	HTTP/1.1

Sample Request/Response

Request Headers

```

GET https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/221c2fa3-6440-4a7a-b1e4-
88a3c8878233/AuthorizationToken?KeyId=f6eb7a33-e745-4e0b-bebb-54abcb56ee5f HTTP/1.1
Content-Type: application/json

```

Query String Parameter

```

KeyId=<"ContentProtectionKeyId">

```



If successful, a **200 OK** status code is returned.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 591
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://azuremediaservicesconnectorc0691ddb6f2424ea4aa3e4644091464.azurewebsites.net/api/Videos/221c2f
a3-6440-4a7a-b1e4-88a3c8878233/AuthorizationToken?KeyId=f6eb7a33-e745-4e0b-bebb-54abc56ee5f
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 21:19:30 GMT

{
  "value": "Bearer=urn%3amicrosoft%3aazure%3amediaservices%3acontentkeyidentifier=f6eb7a33-e745-4e0b-
bebb-
54abc56ee5f&urn%3amicrosoft%3aazure%3amediaservices%3akeyacquisitionhostname=cvprbl203m01.keydelivery.
mediaservices.windows.net&http%3a%2f%2fschemas.microsoft.com%2faccesscontrolservice%2f2010%2f07%2fclaim
s%2fidentityprovider=https%3a%2f%2fnimbuslkgglobs.accesscontrol.windows.net&urn%3aServiceAccessible=s
ervice&Audience=urn%3aNimbus&ExpiresOn=1427145871&Issuer=https%3a%2f%2fnimbuslkgglobs.accesscontrol.w
indows.net%2f&HMACSHA256=rjG0atj1iIavspn%2b894DtHFpZM4dWXL21ULgB26QeEo%3d"
}
```

2.3.2.2.6 CHECK STATUS

Clients use this to check if the content is ready for streaming

Method	Request URI	HTTP Version
GET	GET /api/Videos/{id}/Status	HTTP/1.1

Sample Request/Response

Request Headers

```
GET https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/221c2fa3-6440-4a7a-b1e4-
88a3c8878233/Status HTTP/1.1
Content-Type: application/json
```

If the video is still being processed, clients will get a **200 OK** status code.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 99
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/221c2fa3-6440-4a7a-b1e4-
88a3c8878233/Status
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 21:13:23 GMT

{
  "StatusCode":1,
}
```



```

"Status": "Processing",
"Message": "Video is still being processed. Try again later."
}

```

If the video is successfully encoded, clients will get a 200 OK status code with the following status message in the body:

```

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 65
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gateway.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/221c2fa3-6440-4a7a-b1e4-88a3c8878233/Status
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=add8315422140763770796c2c64f39117104178581c607b765efa5f51854d152;Path=/;Domain=ctest2gateway.azurewebsites.net
Date: Mon, 23 Mar 2015 21:17:30 GMT

{
  "StatusCode": 2,
  "Status": "Completed",
  "Message": "Video is ready."
}

```

2.3.3 THUMBNAILS

The Thumbnail entity represents a thumbnail image derived from a video. Its entity path is Videos/{id}/Thumbnails.

2.3.3.1 THUMBNAIL PROPERTIES

Property	Type	Description	Required?
Height	int	Height of thumbnail in pixels	
ID	String	Primary key of entity	
Index	int	For indexed thumbnails, the index of this thumbnail. 0 otherwise.	
Time	int	For time-based thumbnails, the time offset in seconds.. 0 otherwise.	
Url	string	The URL of the thumbnail. The URL returned here is valid for a limited period of time.	
Videoid	String	The key of the Video from which this Thumbnail is derived.	

2.3.3.2 SUPPORTED OPERATIONS

2.3.3.2.1 GET A THUMBNAIL

Method	Request URI	HTTP Version
GET	/api/Videos/{id}/Thumbnails	HTTP/1.1

Sample Request

Request Headers

```

GET https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/4f13e32f-7eab-4c24-948a-f817647dbe2b/Thumbnails HTTP/1.1
Content-Type: application/json

```

Request Body

None



If successful, a **200 OK** status code is returned along with a representation of the created entity in the response body.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 520
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/9457d368-3a6b-4e8a-a64e-alec5e3e357b/Thumbnails
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
Date: Mon, 23 Mar 2015 19:57:19 GMT

[
  {
    "Kind": "Poster",
    "Height": "960",
    "Index": "0",
    "Time": "0",

    "Url": "https://cvprbl203v.cloudvideo.azure.net:443/api/ImageProxy?mediaservicesurl=https%3a%2f%2fcvprbl203m01r14.blob.core.windows.net%2fasset-e41e435d-1500-80c3-01a7-f1e4d195f969%2fPoster.jpg%3fsv%3d2012-02-12%26sr%3dc%26si%3d37390e8f-c4d2-4125-8233-df100fa99b84%26sig%3dpjvZnvGMDEctg3bc1ekKaMtASELYjtaKeilXhwODIoM%253D%26st%3d2015-03-23T19%253A47%253A30Z%26se%3d2015-04-06T19%253A47%253A30Z&imagemetadata=nb%3acid%3aUUID%3ae41e435d-1500-80c3-01a7-f1e4d195f969"
  }
]
```

If a wrong video id is specified, you will get a **404 – Not Found**.

2.3.4 PLAYBACK

The Playback action is used to retrieve the playback URL of a Video. Its entity path is PlaybackEndpoint. This PlaybackEndpoint will only be available after the video is ready for playback and all processing is completed in the backend.

2.3.4.1 SUPPORTED OPERATIONS

This endpoint supports the GET operation with the video id parameter. This call also requires query parameter of “endpointType”, with values as shown in the table below.

2.3.4.1.1 GET PLAYBACK URL

Method	Request URI	HTTP Version
GET	/api/Videos/{id}/PlaybackEndpoint?endpointType= {‘endpoint type’}	HTTP/1.1

Where, endpointType can be one of the following:

endpointType value	Playback format
Streaming	Microsoft Smooth Streaming
HLS	Apple HTTP Live Streaming (HLS) format
DASH	Dynamic Adaptive Streaming over HTTP (DASH), also known as MPEG-DASH
Progressive	H.264 Progressive download

Sample Request/Response



Request Headers

```
GET https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/9457d368-3a6b-4e8a-a64e-alec5e3e357b/PlaybackEndpoint?endpointType=Streaming HTTP/1.1
```

Request Body

None

If successful, a **200 OK** status code is returned along with a representation of the created entity in the response body.

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 183
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://testazuremediaservicesconnector.azurewebsites.net/api/Videos/9457d368-3a6b-4e8a-a64e-alec5e3e357b/PlaybackEndpoint?endpointType=Streaming
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gateway.azurewebsites.net
Date: Mon, 23 Mar 2015 19:52:42 GMT

{
  "Type": "Streaming",
  "PlaybackUrl": "http://cdn-cvprbl203m01.streaming.mediaservices.windows.net/b8400467-959d-4707-b418-2906b2574460/9dcf086e-8866-44fd-83fa-6888913e9986.ism/Manifest"
}
```

If the VideoID is invalid, you will get a **404 Not Found** status.

If the video is still processing you will get **403 – Forbidden** status.

2.3.5 FORMATS

The Formats endpoint is used to retrieve a list of supported input video formats. Its entity path is Formats.

2.3.5.1 PROPERTIES

Property	Type	Description	Required?
Extension	String	File extension of format	
MimeType	String	The MIME type for this format	

2.3.5.2 SUPPORTED OPERATIONS

This endpoint supports the GET operation.

2.3.5.2.1 GET FORMATS

Method	Request URI	HTTP Version
GET	/api/Formats	HTTP/1.1

Sample Request/Response

Request Headers

```
GET https://testazuremediaservicesconnector.azurewebsites.net/api/Formats HTTP/1.1
```

Request Body

None

If successful, a **200 OK** status code is returned along with a representation of the created entity in the response body.



```

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Length: 1746
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
x-ms-proxy-outgoing-newurl:
https://azuremediaservicesconnectorc0691ddb6f2424ea4aa3e4644091464.azurewebsites.net/api/Formats
X-Powered-By: ASP.NET
Set-Cookie:
ARRAffinity=4ffc8e980cb737c94f36786019b7991cf401d053cc900e4e68629f786f0d288b;Path=/;Domain=ctest2gatewa
y.azurewebsites.net
Date: Mon, 23 Mar 2015 20:07:35 GMT

```

```

[{"Extension": "aif", "MimeType": "audio/x-aiff"}, {"Extension": "aifc", "MimeType": "audio/x-
aiff"}, {"Extension": "aiff", "MimeType": "audio/x-
aiff"}, {"Extension": "au", "MimeType": "audio/basic"}, {"Extension": "avi", "MimeType": "video/x-
msvideo"}, {"Extension": "dif", "MimeType": "video/x-dv"}, {"Extension": "dv", "MimeType": "video/x-
dv"}, {"Extension": "m3u", "MimeType": "audio/x-mpegurl"}, {"Extension": "m4a", "MimeType": "audio/mp4a-
latm"}, {"Extension": "m4b", "MimeType": "audio/mp4a-latm"}, {"Extension": "m4p", "MimeType": "audio/mp4a-
latm"}, {"Extension": "m4u", "MimeType": "video/vnd.mpegurl"}, {"Extension": "m4v", "MimeType": "video/x-
m4v"}, {"Extension": "mid", "MimeType": "audio/midi"}, {"Extension": "midi", "MimeType": "audio/midi"}, {"Extens
ion": "mov", "MimeType": "video/quicktime"}, {"Extension": "movie", "MimeType": "video/x-sgi-
movie"}, {"Extension": "mp2", "MimeType": "audio/mpeg"}, {"Extension": "mp3", "MimeType": "audio/mpeg"}, {"Exten
sion": "mp4", "MimeType": "audio/mp4"}, {"Extension": "mpe", "MimeType": "audio/mpeg"}, {"Extension": "mpeg", "Mi
meType": "audio/mpeg"}, {"Extension": "mpg", "MimeType": "audio/mpeg"}, {"Extension": "mpga", "MimeType": "audio
/mpeg"}, {"Extension": "mxu", "MimeType": "video/vnd.mpegurl"}, {"Extension": "qt", "MimeType": "video/quicktim
e"}, {"Extension": "ra", "MimeType": "audio/x-pn-realaudio"}, {"Extension": "ram", "MimeType": "audio/x-pn-
realaudio"}, {"Extension": "wav", "MimeType": "audio/x-wav"}, {"Extension": "asf", "MimeType": "video/x-ms-
asf"}, {"Extension": "asx", "MimeType": "video/x-ms-asf"}, {"Extension": "wma", "MimeType": "audio/x-ms-
wma"}, {"Extension": "wax", "MimeType": "audio/x-ms-wax"}, {"Extension": "wmv", "MimeType": "audio/x-ms-
wmv"}, {"Extension": "wvx", "MimeType": "video/x-ms-wvx"}, {"Extension": "wm", "MimeType": "video/x-ms-
wm"}, {"Extension": "wmx", "MimeType": "video/x-ms-wmx"}]

```

2.4 UPLOADING ENCRYPTED CONTENT

For clients that require encryption for stored video, Media Services allows encryption of the input video. This section describes how input video is encrypted.

Clients specify input video encryption by setting the `InputStorageEncryptionMode` property of the `Video` to `AES256`. On video create, Media Services Connector then assigns a symmetric key and initialization vector (IV) for use by the client.

The client is then responsible for encrypting the content using the AES-256 algorithm. Use cipher mode CTR and no padding. C# clients can use the `AesCryptoServiceProvider` class to perform the encryption.

2.5 UPLOAD VIDEO TO AZURE BLOB

Refer MSDN documentation [here](#) for [Sample JavaScript Code for uploading a Blob to Azure Storage](#)

3 END-2-END WORKFLOW

3.1 SCENARIOS

3.1.1 (UNENCRYPTED) CONTENT CREATION AND PLAYBACK SCENARIO

- Create Video
- Upload Video to Azure BLOB
- Indicate Completion of Upload
- Wait for Encoding Completion



- Get Playback URL for the desired playback format
- Get Thumbnail URL for the image to be displayed
- Configure [Azure Media Services Player](#) or your own player to playback media.

3.1.2 ENCRYPTED CONTENT CREATION AND PLAYBACK SCENARIO

- Create Video with Encryption option selected.
- Upload Video to Azure BLOB
- Indicate Completion of Upload
- Wait for Encoding Completion
- Get Authorization Token to playback encrypted content (you will have to refresh the video entity to get the ContentProtectionKeyId)
- Get Playback URL for the desired playback format
- Get Thumbnail URL for the image to be displayed
- Configure [Azure Media Services Player](#) or your own player to playback media.

4 REPORTING ISSUES

When reporting issues with the service, a sample request and response helps the Media Services team troubleshoot the issue. If you're unable to provide a sample, please provide the data shown in this table:

Troubleshooting Data
URL being requested
HTTP method
Date/Time of request
request-id header

