Haivision Media Platform 2.6

REST API Integrator's Reference

HVS-ID-INT-HMP-API-26, Issue 01

Edition Notice

About Haivision

Founded in 2004, Haivision is now a market leader in enterprise video and video streaming technologies. We help the world's top organizations communicate, collaborate and educate. Recognized as one of the most influential companies in video by Streaming Media and one of the fastest growing companies by Deloitte's Technology Fast 500, organizations big and small rely on Haivision solutions to deliver video. Headquartered in Montreal, Canada, and Chicago, USA, we support our global customers with regional offices located throughout the United States, Europe, Asia and South America.

Trademarks

The Haivision logo, Haivision, and certain other marks are trademarks of Haivision. CoolSign is a registered trademark licensed to Haivision Systems, Inc. All other brand or product names identified in this document are trademarks or registered trademarks of their respective companies or organizations.

Disclaimer

The information contained herein is subject to change without notice. Haivision assumes no responsibility for any damages arising from the use of this content, including but not limited to, lost revenue, lost data, claims by third parties, or other damages.

If you have comments or suggestions, please contact **infodev@haivision.com**.

While every effort has been made to provide accurate and timely information regarding this product and its use, Haivision Systems Inc. shall not be liable for errors or omissions contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Contents

About This Document

Conventions

The following conventions are used to help clarify the content.

Typographic Conventions and Elements

Italics	Used for the introduction of new terminology, for words being used in a different context, and for placeholder or variable text.
bold	Used for strong emphasis and items that you click, such as buttons.
`Monospaced`	Used for code examples, command names, options, responses, error messages, and to indicate text that you enter.
>	In addition to a math symbol, it is used to indicate a submenu. For instance, **File** > **New** where you would select the New option from the File menu.
...	Indicates that text is being omitted for brevity.

Action Alerts

The following alerts are used to advise and counsel that special actions should be taken.

Tip

Indicates highlights, suggestions, or helpful hints.

Note

Indicates a note containing special instructions or information that may apply only in special cases.

Important

Indicates an emphasized note. It provides information that you should be particularly aware of in order to complete a task and that should not be disregarded. This alert is typically used to prevent loss of data.

Caution

Indicates a potentially hazardous situation which, if not avoided, may result in damage to data or equipment. It may also be used to alert against unsafe practices.

Warning

Indicates a potentially hazardous situation that may result in physical harm to the user.

Obtaining Documentation

This document was generated from the Haivision InfoCenter. To ensure you are reading the most up-to-date version of this content, access the documentation online at **https://doc.haivision.com**. You may generate a PDF at any time of the current content. See the footer of the page for the date it was generated.

Getting Service Support

For more information regarding service programs, training courses, or for assistance with your support requirements, contact Haivision Technical Support using our Support Portal at: **https://support.haivision.com**.

This guide describes how to configure, manage, and create custom applications for the Haivision Media Platform (HMP) software using the REST API (Application Programming Interface).

REST API Introduction

The Haivision Media Platform (HMP) REST API is an application programming interface that may be used to create custom applications for HMP.

For an introduction to HMP, as well as information on managing HMP from the Web interface, please see the *HMP User's Guide* and the *HMP Administrator's Guide*.

> ⓘ **Note**
>
> A Command Line API is available as an alternative to the REST API. The Command Line API provides streamlined remote control of core HMP functionality, such as the creation and management of sessions and videos. For more information, see the *HMP Command Line API Integrator's Guide*.

Topics Discussed

Introduction to HMP REST API

The HMP REST API is a Representational State Transfer (REST) Application Programming Interface (REST API). REST is a style of software architecture for distributed hypermedia systems such as the World Wide Web. REST provides a set of rules (constraints) to which an architecture should conform. This is in contrast to an "unconstrained architecture" in which services are free to define their own idiosyncratic interfaces.

REST provides a uniform interface between components, allowing them to communicate in a standard way. Requests use the standard HTTP methods. GET, PUT and DELETE requests can do only what is expected. The effect is that your services are accessible through standard tools, and it is safe for other services and utilities to use yours in ways you did not predict.

REST Informational Links

Following are some useful external references to learn more about REST:

- **Architectural Styles and the Design of Network-based Software Architectures** (dissertation by Roy Fielding)
- **Representational State Transfer** (Wikipedia entry)
- **Explaining REST**
- **How to Create a REST Protocol**
- **REST Anti-Patterns**

Introduction to JSON

The HMP REST API uses JSON (JavaScript Object Notation) to format API requests and responses. JSON is a lightweight data-interchange format that is easy for both humans to read and write, and machines to parse and generate. It is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, and Python.

JSON is built on two universal data structures, which are supported by nearly all modern programming languages:

- A collection of name/value pairs. In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an array, vector, list, or sequence.

A full description of JSON can be found at **http://www.json.org**.

HMP Concepts

This section introduces HMP concepts that you should understand to use the REST API effectively.

- **HMP Resources**
- **Metadata**
- **Permissions**

HMP Resources

The HMP REST API can create and manipulate a variety of resources that reside on the HMP system. This section provides a brief description of their functions.

Identifiers

Each unique HMP object is assigned a randomly generated identifier (ID) when it is created. The REST API uses these IDs to identify specific resources.

Sources

A *source* is an incoming HMP-compatible MPEG-TS video stream that HMP content creators can use to generate a session. See **Source Resources**.

Sessions

A *session* is a collection of one or more video sources that are identified to provide live viewing, recording, and metadata storage. See **Session Resources**.

Recordings

A *recording* is a collection of one or more stored videos for playback and metadata storage.

Recording Type	Description
Active Recording	An active recording is a recording that is currently being created by recording a session's sources. When it is paused, it becomes a paused recording. When it is stopped it becomes a finished recording.
Paused Recording	A paused recording is an active recording that has been suspended. When it is resumed, it becomes an active recording. When it is stopped, it becomes a finished recording.

Finished recordings are referred to as *videos*. A video is a recording that has been finalized and is available for playback in its entirety. It is no longer considered active and it may not be paused or resumed. See **Recording Resources**.

Stream

A *stream* is a HMP-compatible MPEG-TS video stream that is generated by HMP as it re-streams a finished video. See **Stream Resources**.

Metadata

HMP uses metadata to provide meaningful information about sessions and videos. This section provides an overview of the kinds of predefined and custom metadata that HMP supports.

- **Common Metadata**
- **Category Metadata**
- **Metadata Names**
- **Category Metadata Names**
- **Fields**
- **HotMarks**

Common Metadata

HMP stores common metadata for sessions and videos. This type of metadata is defined on all HMP systems (as opposed to Category metadata which is custom and defined by an administrator). The following types of common metadata are used by HMP.

Metadata Type	Description
`title`	Name given to a session or video.
`description`	Free-form text account of a session or video.
`calypso:creator`	Username of the HMP user who created the session or video. See **Session Metadata** and **Video Metadata**.
`calypso:creatorDisplayName`	Display name of the user who created the session or video.

HMP’s common metadata corresponds with the Dublin Core Metadata Element Set:

HMP Common Metadata	Dublin Core Element
`title`	`title`
`description`	`description`
`creator`	`creator`

For more information about the Dublin Core Metadata Element Set refer to IETF RFC 5013.

Category Metadata

HMP administrators can define metadata categories with default values to identify and store custom metadata for sessions and videos. HMP users can assign these categories and values to sessions and videos. Viewers can then select category/value pairs to filter the displayed list of sessions or videos.

The REST API allows developers to query the available categories and predefined values. See Category Resources. All Category administration must be done from the HMP Admin Web interface.

Metadata Names

The HMP `session` and `asset` metadata use a namespace convention. Metadata defined by Haivision will have a name that starts with “`calypso`” or “`haivision`” followed by a “:” and the field name.

The following metadata names are defined. Use caution when modifying Haivision-defined metadata.

- `calypso:creator`
- `calypso:recorded_from_session`
- `calypso:expiration_timestamp`
- Any metadata element with the prefix `calypso:category:`

The `calypso` and `haivision` metadata namespaces are reserved for use by Haivision.

Category Metadata Names

HMP categories are stored as metadata on sessions and videos (`assets`) and use a `calypso:category` namespace. Category keys are used in the metadata entry name; for example, a category with the key “`Room Number`” is stored in the metadata entry named “`calypso:category:Room Number`”.

Fields

Metadata JSON entities consist of an array of Metadata Entries. Each entry has a key and an array containing one or more fields. Each field contains a single typed value. Fields require a type and appropriate value property.

The value of an HMP metadata field can be a string, a 32-bit signed integer, or a 32-bit unsigned integer. Each supported field type has a corresponding JSON value property:

- STRING - valueString
- INT32 - valueInt32

- UINT32 - valueUint32

HMP does not restrict the names of the contents of metadata entries. However, it uses several entries to store specific information:

- `calypso:recorded_from_session` has a STRING field that is the id of the generated session.
- `calypso:creator` has a STRING field that is the username of the HMP user who created the session or video.
- `calypso:creatorDisplayName` has a STRING field that is the display name of the HMP user who created the session or video.
- `calypso:expiration_timestamp` has an INT32 field that contains the time a video should expire and be removed from the system. Time is in seconds since the Unix epoch.
- Metadata elements with the prefix `calypso:category:` are used to store Category values. Each category can have multiple STRING fields.

HotMarks

HMP stores HotMarks that indicate points of interest in a video. Each HotMark is a metadata bookmark that allows viewers to reference points of interest in the video. From the REST API, programmers can add HotMarks to annotate a video (active or completed). See **Create a HotMark on a Recording** and **Video HotMarks**.

Permissions

HMP uses a combination of share permissions and roles with permissions to provide users or groups with controlled access to sessions, videos, and sources. If a user does not have permission to at least view a session, video, or source, it does not appear in the portal. From the REST API, programmers can assign these permissions to users or groups.

User can customize share permissions by granting and revoking specific actions on any available share permission. HMP comes with the default pre-configured share permissions listed below.

For details on share permissions and how to customize them, see "**Managing Roles**" in the *Haivision Media Platform Administrator's Guide*.

Session Default Share Permissions

Permission Type	**Description**
OWN	The default OWN permission grants the rights to watch, edit, share, record, change sources, and delete a session.
EDIT	The default EDIT permission grants the rights to watch and edit a session.
WATCH	The default WATCH permission grants the right to watch a session.

See **Session Share Permissions**.

Video Default Share Permissions

Permission Type	**Description**
OWN	The default OWN permission grants the rights to watch, edit, share, re-stream, trim, download, and delete a video.

Permission Type	Description
EDIT	The default EDIT permission grants the rights to watch and edit a video.
WATCH	The default WATCH permission grants the right to watch a video.

See Video Share Permissions.

Source Default Share Permissions

Permission Type	Description
OWN	The default OWN permission grants the rights to edit, share, and watch a source.
EDIT	The default EDIT permission grants the rights to edit and watch a source.
WATCH	The default WATCH permission grants the right to watch a source

See Source Share Permissions.

Default Roles

Roles are used to confer permissions to users and groups. A user must be assigned a role to log in. HMP provides the following four default roles:

Role	Permissions
Administrator	In charge of system
Content Creator	Make sessions and create videos
Content Contributor	Create videos with no other system responsibilities
Viewer	View or interact with content with no other system responsibilities
STB	Same as Content Creator with Set-Top Box administration added

For details on the default role permissions, as well as customizing roles, see "Managing Roles" in the *Haivision Media Platform User's Guide*.

REST API Concepts

This section introduces REST API concepts underlying the HMP REST API.

- Required Authorization
- Data and Paging Attributes
- Default Success HTTP Status Code
- JSON Properties
- HTTP Request Headers
- Queries
- Sorting
- Errors

Required Authorization

All REST API requests must be sent over the secure HTTPS protocol. Also, each REST API implementation needs to authenticate with valid user credentials to obtain a session cookie. To get a session cookie, execute an **Initiate User Session** command with your credentials. The response includes a "`calypso-session-id`" cookie. All API requests must be made with a valid session cookie in the request header. Web browsers and browser-based REST tools should handle cookie management automatically. Browser-less interactions may require cookie headers to be manually managed.

To terminate the session, execute a **Delete a User Session**. You can check if a session exists or is valid with a **Get User Session Info** command. See **Authentication Resources** for more details on these commands.

Also, each API command listed in **REST API Reference** includes an "Authorizations" heading that lists the permission necessary to execute the command. If you make an API call using a login that is not authorized to execute the command, a 403 Forbidden error response is returned, which is described in **Errors**.

Data and Paging Attributes

All APIs are expected to wrap their resource and collection data in a `data` attribute. Resources have a single object under data, and collections have an array of objects.

Collections are paged and have a `paging` attribute in the response, in addition to data. When there is more paged data available, a `next` attribute is present on paging with a link to the next data set. If no more data is available, the next attribute is not part of the paging response. Alternatively, `page` and `pageSize` parameters can be used to access paged data. For example, `GET /apis/assets?page=2&pageSize=100` tries and fetch videos (`assets`) 101-200 from the collection.

For example:

```
{
    "data": [
        {
            "title": "InfoDevSession - 18/03/2014, 15:06",
            "description": "API Test",
            "active": false,
            "id": "3bc946a6-6720-41c0-8246-55e8119d1a5a",
            "ctime": 1401997809,
            "mtime": 1402002014,
            "tag": null,
            "duration": 60.347000000000001,
            "recording": false,
            "trimming": false,
            "importing": false,
            "movieTrackCount": 1
        }
    ],
    "paging": {
        "results": 3,
        "pageSize": 1,
        "next": "https:/calypso.example.com/apis/assets?page=2&pageSize=1"
    }
}
```

Default Success HTTP Status Code

Unless otherwise specified, the HTTP status codes for success responses are as follows:

HTTP Method	HTTP Status Code
GET	200
POST	201
PUT	200
DELETE	200

JSON Properties

JSON properties are either optional, required, immutable, or ignored. The definitions are as follows:

Property	Description
Optional	If the property is not provided on a PUT or POST, the server uses the default value for PUT requests or the current value stored on the server for POST. (Example: the description of a session).
Required	If the property is not provided on a PUT or POST, an error is returned stating that the property must be provided. (Example: the timestamp of a HotMark).
Ignored	Ignored properties are informational only in responses and are ignored for PUT and POST requests.
Immutable	Immutable properties cannot be changed. If a PUT or POST request attempts to change or assign an immutable property, an error is returned.

HTTP Request Headers

On a POST or PUT request, the content type request header should be one of the content types supported by HMP for that specific API endpoint, either:

- `application/json`
- `application/octet-stream`

If the content type is different, the HTTP error `415 Unsupported Media Type` is returned.

Queries

You can query sources, sessions, and videos based on a title and description (or HotMark title and description, where applicable) by adding the "q" query parameter to the collection URL.

Queries are case-insensitive and search title, description, `calypso:creator`, and HotMarks. For example: `GET /apis/assets?q="My Search Term"`

You can add filters for filtering categories. The filter format is as follows:

`category::"label1" IN ("value1", "value2", ..., "valueN") & category::"label2" IN ("value1", "value2", ..., "valueM") & ...`
Example:
`category::"Patient ID" IN ("123", "456", "First", "Second") & category::"Doctor" IN ("john", "Bob")`

`category::"label1" NOT IN ("value1", "value2", ..., "valueN") & ...`
Example:
`category::"Patient ID" NOT IN ("123", "456", "First", "Second") & category::"Doctor" IN ("john", "Bob")`

`category::"label1" IN ("") & ...`
Example:
`category::"Patient ID" IN ("")`

`category::"label1" NOT IN ("") & ...`
Example:
`category::"Patient ID" NOT IN ("")`

Operators:

- `IN ("value1", "value2", ..., "valueN")` – case-insensitive values in the set are ORed and the results are returned.
- `NOT IN ("value1", "value2", ..., "valueN")` – case-insensitive values in the set are ORed and the negated results are returned.
- `"category" IN ("")` – returns all results that have this category regardless of value.
- `"category" NOT IN ("")` – returns all results that do not have this category.
- `&` – Binary operand values are ANDed.

Label and value must be quoted. Label and value string must have special characters escaped (based on *ECMA 262*, 5th edition, Section 7.8.4 "String Literals").

The following special characters must be escaped:

```
"  -> \\"
\\ -> \\\\
\b -> \\b
\t -> \\t
\n -> \\n
\v -> \\v
\f -> \\f
\r -> \\r
```

For example, you can use JSON.stringify() to do the escaping. You need to encode the filter string with urlencode and append the encoded string after the "filter" parameter in the URL.

Example:

```
GET /apis/assets?q="My Search Term"&filter= category%3A%3A%22Patient+ID%22+IN+%28123%2C+456%2C+
%22First%2 2%2C+%22Second%22%29+%26+category%3A%3ADoctor+IN+%28%22john%22%2C+%22Bob%22%29
```

Sorting

With sessions and videos, you can sort the API response using the `sort` and `asc` parameters.

- To perform an ascending sort, use `asc=1`. *This is the default sort.*
- To perform a descending sort, use `asc=-1`.

The following sort fields can be specified:

- title
- description
- duration
- ctime

For example:

```
GET /apis/assets?sort=title&asc=-1
```

Errors

In case of an error on an API call, the API returns a response with the appropriate HTTP status code and an error message encapsulated in the body:

```
{
  "code": "xxxxxx",
  "message": "Error message here...",
  "httpStatusCode": "xxx",
  "name": "yyy"
}
```

Errors codes are grouped by status codes:

Error Code	HTTP Status Code
00xxxx	Command Line API errors, Unknown errors
01xxxx	400 Bad Request
02xxxx	401 Not Authorized
03xxxx	403 Forbidden
04xxxx	404 Not Found
05xxxx	405 Method Not Allowed
06xxxx	409 Conflict
07xxxx	500 Internal Server Error
08xxxx	501 Not Implemented
09xxxx	503 Service Unavailable
100000	415 Unsupported Media Type

REST API Reference

This API command reference lists and describes the available resources for the HMP REST API. This is a list of API calls that can be used to create and manage sessions and record and share video on an HMP system. For conceptual and setup information, see REST API Introduction.

Topics Discussed

- Summary of HMP REST API Resources
- Authentication Resources
- User Resources
- Group Resources
- Source Resources
- Session Resources
- Recording Resources
- Category Resources
- Asset Resources
- Stream Resources
- System Resources
- Event Resources

Summary of HMP REST API Resources

REST API Endpoint	Method	Use to...
/apis/authentication/login		
	GET	Get User Session Info
	POST	Initiate User Session
	DELETE	Delete a User Session
/apis/users/enabled		
	GET	Get the User List
/apis/groups/enabled		
	GET	Get the Group List
/apis/sources		
	GET	Get the Source List
	POST	Create a Source
/apis/sources/:sourceId		
	GET	Get Information on a Source
	PUT	Edit a Source

REST API Endpoint	Method	Use to...
	DELETE	Delete a Source
/apis/sources/:sourceId/metadata		
	GET	Get a List of Metadata
	PUT	Edit Source Metadata
	DELETE	Delete Source Metadata
/apis/sources/:sourceId/shares		
	GET	View Sharing Permissions for a Source
	POST	Create Share Permissions for a Source
/apis/sources/:sourceId/shares/:shareId		
	GET	Get Share Permission Information
	PUT	Edit Share Permissions
	DELETE	Delete Share Permissions
/apis/sources/:sourceId/thumbnail		
	GET	Get a Video Thumbnail
/apis/sessions		
	GET	Get a List of Sessions
	POST	Create a Session
/apis/sessions/:sessionId		
	GET	Get Session Information
	PUT	Edit a Session
	DELETE	Delete a Session
/apis/sessions/:sessionId/assets		
	GET	Get a List of Videos from a Session
/apis/sessions/:sessionId/metadata		
	GET	Get a List of Session Metadata
	PUT	Edit Session Metadata
	DELETE	Delete Session Metadata
/apis/sessions/:sessionId/shares		
	GET	Get Share Permissions for a Session
	POST	Create Share Permissions for a Session
/apis/sessions/:sessionId/shares/:shareId		
	GET	Get Share Permission Information for a Session
	PUT	Edit Share Permission for a Session
	DELETE	Delete Share Permission for a Session

REST API Endpoint	Method	Use to...
/apis/sessions/:sessionId/sources		
	GET	List Session Sources
	POST	Add a Session Source
/apis/sessions/:sessionId/sources/:sourceId		
	DELETE	Delete Session Sources
/apis/sessions/:sessionId/recordings		
	GET	Get a List of Active Recordings for a Session
	POST	Create a New Recording Session
/apis/sessions/:sessionId/thumbnail		
	GET	Get Thumbnail Image for a Session
/apis/sessions/:sessionId/publicshare		
	GET	Get a List of Public Shares
	POST	Create a Public Share
	DELETE	Delete Public Share
/apis/sessions/:sessionId/events		
	GET	Get an Event List
	POST	Create a Session Event
/apis/sessions/:sessionId/events/:eventId		
	GET	Get Event Information for a Session
	PUT	Edit a Session Event
/apis/recordings		
	GET	Get a List of Active Recordings
/apis/recordings/:recordingId		
	GET	Get Recording Details
	PUT	Pause or Resume a Recording
	DELETE	Stop a Recording
/apis/recordings/:recordingId/hotmarks		
	POST	Create a HotMark on a Recording
/apis/categories		
	GET	Get a List of Categories
/apis/categories/:id		
	GET	Get Category Details
/apis/assets		
	GET	Get a List of Recorded Videos

REST API Endpoint	Method	Use to...
/apis/assets/:id		
	GET	Get Video Details
	PUT	Update a Title or Description
	DELETE	Delete a Video
/apis/assets/:id/metadata		
	GET	Get Video Metadata
	PUT	Edit Video Metadata
	DELETE	Delete Video Metadata
/apis/assets/:id/related		
	GET	Get List of Related Videos
/apis/assets/:id/shares		
	GET	Get Video Sharing Permission Information
	POST	Create Video Share Permissions
/apis/assets/:id/shares/:shareId		
	GET	Get Video Share Permissions
	PUT	Edit Video Share Permissions
	DELETE	Delete Video Share Permissions
/apis/assets/:id/hotmarks		
	GET	Get a List of HotMarks
	POST	Create a New HotMark
/apis/assets/:id/hotmarks/:hotmarkId		
	GET	Get Specific HotMark Information
	PUT	Update a Hotmark
	DELETE	Delete a HotMark
/apis/assets/:id/download		
	GET	Download Videos
/apis/assets/:id/streams		
	GET	Get a Stream List Started from a Video
	POST	Create a Stream
/apis/assets/:id/streams/:streamId		
	GET	Get Stream Information for a Video
	DELETE	Delete a Stream for a Video
/apis/assets/:id/thumbnail		
	GET	Get a Thumbnail Image for a Video

REST API Endpoint	Method	Use to...
	PUT	Update the Thumbnail for a Video
/apis/assets/:id/tracks/:trackId/thumbnail		
	GET	Get a Thumbnail Image for a Video Track
/apis/assets/:id/publicshare		
	GET	Get a Public Share List for a Video
	POST	Create a Public Share for a Video
	DELETE	Delete a Public Share for a Video
/apis/streams		
	GET	Get a Stream List Started from a Video
/apis/streams/:streamId		
	GET	Get Stream Information
	PUT	Pause or Resume a Stream
	DELETE	Stop a Stream
/apis/streams/:streamId/seek		
	POST	Jump to a Point in a Stream
/apis/system/version		
	GET	Get the HMP Software/Build Version
/apis/events		
	GET	Get List of Events

Authentication Resources

Synopsis

The authentication API is used to manage HMP user sessions. Authorization for REST API resources is handled through session cookies.

- To get a session cookie, execute an **Initiate User Session** command with your credentials.
- To terminate the session, execute a **Delete a User Session** command.

You can check if a session exists or is valid with a **Get User Session Info** command.

Authentication Parameters

Property	JSON Type	Property Type	Description
username	string	Required	User name for HMP.
password	string	Required	Password for HMP.

Get User Session Info

Gets information about the user session for the `calypso-session-id` cookie included in the request headers.

Authorizations: All

Requests

```
GET /apis/authentication/login
```

Response

Example:

```
{
  "data": {
    "id": null,
    "sessionId": "b34f71d6-d0b2-470a-9aeb-43ba7a3bb831",
    "username": "haiadmin",
    "admin": true,
    "groups": [],
    "roleIds": [],
    "roles": [
      {
        "id": "admin",
        "name": "Administrator",
        "admin": true,
        "permissions": {}
      }
    ],
    "sessions": {},
    "assets": {},
    "ldapUser": false,
    "displayName": "haiadmin",
    "expires": "2014-05-01T17:50:24.019Z"
  }
}
```

⚠ **Caution**

This data is informational only and subject to change. The User Session resource is not a defined resource with full support in HMP v2.5.

REST API developers are encouraged to check for a `200 OK` status to verify that a session is valid, but otherwise ignore this response. REST API applications that use the data from this response may stop functioning and break user logins if this response is changed.

Error

Code	Payload	Description

`401`	`{"code":"020001",`
	`"name":"UserNotAuthorized",`
	`"message":"User is not authorized",`
	`"httpStatusCode":401}`
		User is not authenticated.

Initiate User Session

This request authenticates the user.

Authorizations: All

Requests

```
POST /apis/authentication/login

{
  "username": "<username>",
  "password": "<password>"
}
```

Response

Example:

```
{
  "data": {
    "id": null,
    "sessionId": "15a3c22d-2ab3-4d3e-a248-8e194f01585e",
    "username": "haiadmin",
    "admin": true,
    "groups": [],
    "roleIds": [],
    "roles": [
      {
        "id": "admin",
        "name": "Administrator",
        "admin": true,
        "permissions": {}
      }
    ],
    "sessions": {},
    "assets": {},
    "ldapUser": false,
    "displayName": "haiadmin",
    "expires": "2014-05-09T18:15:23.380Z"
  }
}
```

> ⚠ **Caution**
>
> This data is informational only and subject to change in future HMP releases. The User Session resource is a private resource in HMP v2.5.

REST API developers are encouraged to check for a `200 OK` status to verify that a session is valid, but otherwise ignore this response. REST API applications that use the data from this response may stop functioning and break user logins if this response is changed.

A `calypso-session-id` cookie is returned via the Set-Cookie HTTP header. This cookie is used to authenticate the user for all subsequent requests.

Error

Code	Payload	Description
400	{"code":"010001","name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	A user name or password was missing or the wrong JSON type.
401	{"code":"020001","name":"UserNotAuthorized", "message":"User has no roles assigned", "httpStatusCode":401}	User has no role assigned.
401	{"code":"020002","name":"InvalidCredentials", "message":"Invalid credentials", "httpStatusCode":401}	The user name and or the password was incorrect.
500	{"code":"070051","name":"CreateUserSessi on", "message":"Could not create user session", "httpStatusCode":500}	An internal error occurred.
500	{"code":"070000","name":"InternalServer", "message":<Error message>, "httpStatusCode":500}	An internal error occurred.
503	{"code":"090002","name":"ServiceUnavailable", "message":"No connection to authentication server", "httpStatusCode":503}	Authentication service is unavailable. Please try again later.

Delete a User Session

Deletes the user session for the `calypso-session-id` cookie included in the request headers.

Authorizations: All

Requests

```
DELETE /apis/authentication/login
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040022","name":"AuthSessionNotFound", "message":"Unknown session", "httpStatusCode":404}`	Authentication session does not exist or has already expired.

User Resources

Synopsis

The users REST API is used to get a list of user accounts enabled for HMP access. Depending on the system configuration, this may be a list of users from the HMP internal database, or a list of users from the linked AD or LDAP Directory Server. A user is considered to have access to HMP when there is a role associated with the user or one of the groups to which it belongs. The users REST API can be used to get the user name or user id to share a video or session with. See Create Session Share Permissons and Create Video Share Permissions.

Note

When the dn, username, displayName, and email attributes are sourced from a remote directory server, they are read-only properties of the HMP users resource. Data is never written back to the remote directory server.

User Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the resource, generated by HMP.
username	String	Immutable	The user's login name, e.g., bsmith.
displayName	String	Ignored	The user’s friendly name, e.g., Bob Smith.
email	String	Ignored	The user's email address.

Get the User List

Gets a list of users that have a role or that belong to a group with a role.

Authorizations: All

Requests

```
GET /apis/users/enabled
```

Response

Success

Array of users that have a role or that belong to a group with a role. Example:

```
{
  "data": [
    {
      "assets": [],
      "displayName": "contrib",
      "dn": "cn=contrib,dccalypso,dc=haivision,dc=com",
      "email": "",
      "id": "b70df497-baf3-4512-a2eb-25fd51f87a35",
      "roles": [
        "<roleID>"
      ],
      "sessions": [],
      "shares": {
        "assets": {
          "2f02b038-2d99-442d-b9d7-629459f9449b": {
            "mode": "EDIT"
          }
        },
        "sessions": {
          "0fb77a60-f7af-4c54-bf27-38e0a36924a1": {
            "mode": "OWN"
          },
          "2f8dce6d-fd8b-459e-890f-97965c88df40": {
            "mode": "WATCH"
          }
        }
      },
      "userAttribute": "cn",
      "userContext": "dc=calypso,dc=haivision,dc=com",
      "userObjectClass": "(objectClass=person)",
      "username": "contrib"
    },
    {
      "assets": [],
      "displayName": "InfoDev",
      "dn": "cn=infodev,dc=calypso,dc=haivision,dc=com",
      "email": "infodev@ haivision.com
                  ",
      "id": "4ef8ffa6-231e-41e1-aaa3-57cc4743668e",

      "roles": [
        "<roleID>"
      ],
      "sessions": [],
      "shares": {
        "assets": {
          "523bbda2-e7bc-4e97-b622-e461938c9a03": {
            "mode": "EDIT"
          }
        }
      },
      "userAttribute": "cn",
      "userContext": "dc=calypso,dc=haivision,dc=com",
      "userObjectClass": "(objectClass=person)",
      "username": "infodev"
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Caution

Only the `id`, `username`, `displayName`, and `email` properties are supported. Other attributes are subject to change and should not be used in a production application.

Error

Code	Payload	Description

404	`{"code":"040012","name":"NoResults",`
	` "message":"No results",`
	`"httpStatusCode":404}`
		No users available.

500	`{"code":"070000","name":"InternalServer",`
	` "message":<Error message>,`
	`"httpStatusCode":500}`
		Error retrieving users.

Group Resources

Synopsis

The groups API is used to get information about groups defined for HMP. Groups are only available if HMP is connected to a Directory Services server. The groups REST API is used to get a list of groups enabled for HMP access. This is the list of groups from the linked Directory Services server. Members of a group are considered to have access to HMP when there is a role associated with the group.

The groups REST API can be used to get the groupname or group id to share a video or session with. See **Create Share Permissions for a Session** and **Create Video Share Permissions**.

Note

The `dn` and `groupname` properties are READ-ONLY properties of the HMP groups resource. Data is never written back to the remote directory server.

Group Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the resource, generated by HMP.
groupname	String	Immutable	The group name.

Get the Group List

Gets a list of groups defined for HMP that have a role.

Authorizations: All

Requests

```
GET /apis/groups/enabled
```

Response

Success

Array of group resources. Example:

```
{
  "data": [
    {
      "baseDn": "dc=simple,dc=haivision,dc=local",
      "dn": "CN=Account Operators,CN=Builtin,DC=simple,DC= haivision,DC=local",
      "groupname": "Account Operators",
      "id": "99db0933-0941-4083-856c-320fadca834e",
      "roles": [
        "<roleID>"
      ],
      "shares": {
        "assets": {
          "05b74e69-4b27-4540-bb0a-c7042e24ce3e": {
            "mode": "OWN"
          },
          "523bbda2-e7bc-4e97-b622-e461938c9a03": {
            "mode": "WATCH"
          }
        },
        "sessions": {
          "0fb77a60-f7af-4c54-bf27-38e0a36924a1": {
            "mode": "WATCH"
          }
        }
      }
    },
    {
      "baseDn": "dc=simple,dc=haivision,dc=local",
      "dn": "CN=Administrators,CN=Builtin,DC=simple,DC=haivision,
               DC=local",
      "groupname": "Administrators",
      "id": "b5b45f60-0962-4bed-8ec1-a26f76c0a1ea",
      "roles": [
        "<roleID>"
      ]
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Caution

Only `id` and `groupname` are supported. Other attributes are subject to change and should not be used in a production application.

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	No groups available.
500	`{"code":"070000", "name":"InternalServer", "message":"Internal Server Error", "httpStatusCode":500}`	Error retrieving groups.

Source Resources

Synopsis

A source is an incoming unicast or multicast MPEG-TS video stream that can be viewed or recorded after it has been added to a Session. The sources API is used to create and manage the streaming A/V sources to be available for content creators to view and capture.

Source Parameters

Property	JSON Type	Property Type	Description
name	String	Required	Source name.
type	String	Required	Source type, either UDP or SRT.
host	String	Required, for multicast UDP, SRT caller, or SRT rendezvous Optional, when multicast is false	Hostname of the source.
port	Number	Required	Port number.
multicast	Boolean	Required if type is UDP	True: source is a multicast stream . False: source is a unicast stream directed to the specified port.
id	String	Immutable	Identifies the source, generated by HMP.
active	Boolean	Ignored	Whether stream data is available.
bitrate	Number	Ignored	Bitrate at which source is sending data.
ctime	Number	Ignored	Source creation time: Unix timestamp (seconds).
mtime	Number	Ignored	Source last modified time: Unix timestamp (seconds).
description	String	Optional	Source description
continuous	Boolean	Optional, value forced to true for SRT	Default is false.
mode	String	Required, if type is SRT	The SRT Connection Mode, either CALLER, LISTENER, or RENDEZVOUS.

Property	JSON Type	Property Type	Description
latency	Number	Optional	Configured latency for SRT in milliseconds.
passphrase	String	Optional	Encrypted SRT passphrase.

Source Actions

- Get the Source List
- Get Information on a Source
- Create a Source
- Edit a Source
- Get a List of Metadata
- Edit Source Metadata
- Delete Source Metadata
- Get a Video Thumbnail
- Delete a Source

Get the Source List

This command get a list of sources created for HMP.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources
```

✓ **Tip**

You can query sources based on a title and description by adding the " `q` " query parameter to the collection URL. For more information, see **Queries**.

Response

Success

```
{
  "data": [
    {
      "description": "",
      "ctime": 1428324787,
      "mtime": 1428324787,
      "host": "10.65.11.174",
      "port": 9942,
      "type": "SRT",
      "continuous": true,
      "mode": "CALLER",
      "passphrase": "",
      "active": true,
      "bitrate": 8442882,
      "id": "77b931f7-5843-4355-886f-45a28c02ab49",
      "name": "Makito SRT"
    },
    {
      "description": "",
      "host": "0.0.0.0",
      "port": 9000,
      "type": "SRT",
      "mode": "LISTENER",
      "latency": 125,
      "passphrase": "passphrase",
      "id": "23f57209-955f-4c73-b9b0-d6dc0aae56a1",
      "continuous": true,
      "ctime": 1427816478,
      "mtime": 1427922685,
      "active": true,
      "bitrate": 224763,
      "name": "Source - 3\/31\/2015, 10:40 AM"
    },
    {
      "description": "InfoDev Test Source",
      "ctime": 1427902484,
      "mtime": 1427902484,
      "host": "239.159.103.100",
      "port": 4900,
      "type": "UDP",
      "multicast": true,
      "continuous": false,
      "active": true,

      "bitrate": 0,
      "id": "5fa26dfd-f2de-4a61-998e-3e27578d7640",
      "name": "Multicast Source"
    }
   ],

  "paging": {
    "results": 3,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	Supplied data was insufficient or incorrect.

Get Information on a Source

Gets information about the specified source.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources/:sourceId
```

Response

Success

```
{
  "data": {
    "name": "InfoDevSource",
    "type": "UDP",
    "host": "239.35.126.101",
    "port": 4902,
    "multicast": true,
    "id": "6e6862e0-d65d-4de5-b690-62e680d4587b"
    "active": true,
    "continuous": false,
    "bitrate": 8345744,
    "ctime":1400879398,
    "mtime":1401023892,
    "description":""
  }
}
```

Error

Code	Payload	Description
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Source with UUID `sourceId` not found.

Create a Source

This method creates a source.

Authorizations: Sources - CREATE

Requests

```
POST /apis/sources

{
  description: ""
  host: "239.35.126.101"
  multicast: true
  name: "InfoDevSource"
  port: 4902
  type: "UDP"

}
```

Response

Success

```
{
  "data": {
    "name": "InfoDevSource",
    "type": "UDP",
    "host": "239.35.126.101",
    "port": 4902,
    "multicast": true,
    "id": "6e6862e0-d65d-4de5-b690-62e680d4587b"
    "active": true,
    "continuous": false,
    "bitrate": 8839739,
    "ctime":1400879398,
    "mtime":1401023892,
    "description":""
  }
}
```

Error

Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

500	`{"code": "070014", "name":"FileWrite",`
	`"message": "Error writing to file",`
	`"httpStatusCode": 500}`
		Unable to write new data to a file.

Edit a Source

Edits the specified source.

Authorizations: Sources - EDIT

Requests

```
PUT /apis/sources/:sourceId

{
  "name": "InfoDevSource2",
  "type": "UDP",
  "host": "239.35.126.101",
  "port": 4902,
  "multicast": true
  "continuous": false,
  "description": "New description"
}
```

Response

Success

```
{
  "data": {
    "name": "InfoDevSource2",
    "type": "UDP",
    "host": "239.35.126.101",
    "port": 4902,
    "multicast": true,
    "id": "6e6862e0-d65d-4de5-b690-62e680d4587b"
    "active": true,
    "bitrate": 7548979,
    "ctime":1400879398,
    "mtime":1402598729,
    "description":"New description"
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect
404	{"code":"040000", "name":"NotFound", "message":"Not Found Error", "httpStatusCode":404}	Unable to locate source.
500	{"code": "070014", "name":"FileWrite", "message": "Error writing to file", "httpStatusCode": 500}	Unable to write updated data to a file.

Get a List of Metadata

Gets a list of metadata for the specified source.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources/:sourceId/metadata
```

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    }
  ]
}
```

Error

Code	Payload	Description
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	The specified source has no metadata.
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Source with UUID `sourceId` not found.

Edit Source Metadata

Edits the metadata for the source.

Authorizations: Sources - EDIT

Requests

```
PUT /apis/sources/:sourceId/metadata

[
  {
  "name": "calypso:creator",
  "field": [
    {
    "fieldType": "STRING",
    "valueString": "haiadmin7"
    }
  ]
  },
  {
  "name": "calypso:category:Locations",
  "field": [
    {
    "fieldType": "STRING",
    "valueString": "Montreal"
    }
  ]
  }
]
```

Parameters

Property	JSON Type	Property Type	Description
name	String	Optional	Name of metadata

field	Array of objects (fields) for these possible structures:
	`{ "fieldType": "STRING", "valueString": "value"}`
	`{ "fieldType": "INT32", "valueInt32": -14}`
	`{ "fieldType": "UINT32", "valueUint32": 42}`
		Optional	Content of metadata. See Fields.

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin7"
        }
      ]
    },
    {
      "name": "calypso:category:Locations",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "Montreal"
        }
      ]
    }
  ]
}
```

Error

Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

404	`{"code":"040009",`
	`"name":"SourceNotFound",`
	`"message":"Source not found",`
	`"httpStatusCode":404}`
		Source with UUID `sourceId` not found.

Delete Source Metadata

Deletes the metadata for the source.

Authorizations: Sources - EDIT

Requests

```
DELETE /apis/sources/:sourceId/metadata
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}`	Source with UUID `sourceId` not found.

Get a Video Thumbnail

Retrieves the thumbnail image for the specified source. Responds with the thumbnail for the first track.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources/:sourceId/shares/:shareId
```

Response

Success

Thumbnail image in JPEG format.

Error

Code	Payload	Description
404	`{"code":"04009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}`	Source with UUID `sourceId` not found.

Delete a Source

Deletes the specified source.

Authorizations: Sources - DELETE

Requests

```
DELETE /apis/sources/:sourceId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Unable to locate source for deletion.
500	{"code":"070014", "name":"FileWrite", "message":"Error writing to file", "httpStatusCode":500}	Unable to write updated data to a file.

Source Share Permissions

- View Sharing Permissions for a Source
- Create Share Permissions for a Source
- Get Share Permission Information
- Edit Share Permissions
- Delete Share Permissions

View Sharing Permissions for a Source

The share resource is used to manage sharing permissions for the source.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources/:sourceId/shares
```

Response

Success

```
{
  "data": [
    {
      "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
      "shareType": "USER",
      "permission": "WATCH",
      "name": "viewer",
      "email": null
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	The specified source has not been shared with any user or group.

Create Share Permissions for a Source

Creates a share permission for a session (i.e., shares the session with the specified user or group).

Authorizations: Sources - SHARE

Requests

```
POST /apis/sources/:sourceId/shares

{
  "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
  "shareType": "USER",
  "permission": "EDIT",
  "name": "viewer"
}
```

Parameters

Property	JSON Type	Property Type	Description
id	UUID	Immutable	Group ID or user ID. Identifies the share entity.
shareType	String	Required Immutable	Accepted values, GROUP or USER: • GROUP: the share grants access to the members of a group. • USER: the share grants access to a specific user.
permission	String	Required	Permission level granted to the share's group or user. Accepted values: WATCH, EDIT, OWN, or <uuid> of admin-defined share permission.
name	String	Optional	Group name or user name.

Note

Either id or name must be present. If both are provided, id is used. See User Resources and Group Resources.

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "EDIT",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Source with UUID `sourceId` not found.
404	{"code":"040004", "name":"UserNotFound", "message":"User not found", "httpStatusCode":404}	The user with provided `id` or `username` was not found.
404	{"code":"040005", "name":"GroupNotFound", "message":"Group not found", "httpStatusCode":404}	The group with provided `id` or `groupname` was not found.
409	{"code":"060001", "name":"UserHasNoRoles", "message":"User has no roles assigned", "httpStatusCode":409}	Make sure a role is assigned to the user or to at least one of the user's groups.
409	{"code":"060002", "name":"GroupHasNoRoles", "message":"Group has no roles assigned", "httpStatusCode":409}	Make sure a role is assigned to the group.
500	{"code":"070008", "name":"FindUser", "message":"Error searching for a user", "httpStatusCode":500}	An internal error occurred while searching for user.
500	{"code":"070009", "name":"FindGroup", "message":"Error searching for a group", "httpStatusCode":500}	An internal error occurred while searching for group.

Get Share Permission Information

Gets information about the specified share permission for the source.

Authorizations: Sources - VIEW

Requests

```
GET /apis/sources/:sourceId/shares/:shareId
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",

    "permission": "EDIT",
    "name": "viewer",
    "email": null
 }
}
```

Error

Code	Payload	Description

404	`{"code":"040021",`
	`"name":"UserGroupNotFound",`
	`"message":"User or Group not found",`
	`"httpStatusCode":404}`
		No user or group was found.

Edit Share Permissions

Edits the specified share permission for the source.

Authorizations: Sources - SHARE

Requests

```
PUT /apis/sources/:sourceId/shares/:shareId

{
  "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
  "shareType": "USER",
  "permission": "WATCH",
  "name": "viewer"
}
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
404	{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	No user or group was found.

Delete Share Permissions

Deletes the specified share permission for the source.

Authorizations: Sources - SHARE

Requests

```
DELETE /apis/sources/:sourceId/shares/:shareId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	No user or group was found.

Session Resources

Synopsis

A session is a collection of one or more video sources that are identified to provide live viewing, recording, and metadata storage. The sessions API is used to create sessions, edit session information, start a recording, and give other users access to the session.

Session Parameters

Property	JSON Type	Property Type	Description
title	String	Required	Title of the session.
description	String	Optional	Description of the session.

Property	JSON Type	Property Type	Description
active	Boolean	Optional	Indicates if the session is active or not. Must be set to true to be available to other users. When false (default), only the session owner can see the session.
syncrecord	Boolean	Optional	If true, indicates the session is recording in synchronous mode for multiple sources (recording all sources by default). Default is false.
movieTrackCount	Number	Ignored	Number of source tracks in the session. (Also, present in the asset resource.)
id	String	Immutable	Identifies the session, generated by HMP.
sources	Array	Ignored	Array of UUIDs corresponding to the source IDs used in the session.

Session Actions

Topics include:

- Create a Session
- Get Session Information
- Edit a Session
- Delete a Session
- Get a List of Videos from a Session
- Get a List of Sessions

Create a Session

Creates a session.

Authorizations: Sessions - CREATE

Requests

```
POST /apis/sessions

{
  "title": "InfoDevSource",
  "description": "UDP",
  "syncrecord": true
}
```

Response

Success

```
{
  "data": {
    "title": "InfoDevSource",
    "description": "UDP",
    "ctime": 1398981914,
    "mtime": 1398981914,
    "duration": 0,
    "active": false,
    "syncrecord": true,
    "id": "3a655931-24b4-4d3d-a897-6e58d361786a",
    "movieTrackCount": 0,
    "sources": [],
    "recording": false
  }
}
```

Error

Code	Payload	Description

`400`	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

`500`	`{"code":"070019",`
	`"name":"CreateSession",`
	`"message":"Failed to create a session",`
	`"httpStatusCode":500}`
		Creating the session failed. Please see `calypso.log` for details.

Get Session Information

Gets information about the specified session.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId
```

Response

Success

```
{
  "data": {
    "title": "ABC",
    "description": "UDP",
    "ctime": 1393533035,
    "mtime": 1398875839,
    "duration": 0,
    "active": false,
    "syncrecord": true,
    "id": "2f8dce6d-fd8b-459e-890f-97965c88df40",
    "movieTrackCount": 1,
    "sources": [
      "f4327f7b-3917-4533-b844-8818741cb0bd"
    ],
    "recording": false
  }
}
```

Error

Code	Payload	Description
403	{"code":"040006", "name":"SessionHidden", "message":"Session hidden", "httpStatusCode":403}	Session is hidden and requester does not own the session.
404	{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}	Session with UUID `sessionId` not found.

Edit a Session

Edits the specified session.

Authorizations: Sessions - EDIT

Requests

```
PUT /apis/sessions/:sessionId

{
  "title": "InfoDevSource2",
  "description": "UDP",
  "syncrecord": true
}
```

Response

Success

```
{
  "data": {
    "title": "InfoDevSource2",
    "description": "UDP",
    "syncrecord": true,
    "active": false,
    "id": "2f8dce6d-fd8b-459e-890f-97965c88df40",
    "ctime": 1393533035,
    "mtime": 1399059859,
    "duration": 0,
    "recording": false,
    "movieTrackCount": 1,
    "sources": [
      "f4327f7b-3917-4533-b844-8818741cb0bd"
    ]
  }
}
```

Error

Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

403	`{"code":"040006",`
	`"name":"SessionHidden",`
	`"message": "Session hidden",`
	`"httpStatusCode":403}`
		Session is hidden and requester does not own the session.

404	`{"code":"040006",`
	`"name":"SessionNotFound",`
	`"message":"Session not found",`
	`"httpStatusCode":404}`
		Session with UUID `sessionId` not found.

500	`{"code":"070000",`
	`"name":"InternalServer",`
	`"message":"Internal Server Error",`
	`"httpStatusCode":500}`
		Updating the session failed. Please see `calypso.log` for more details.

Delete a Session

Deletes the specified session.

Authorizations: Sessions - DELETE

Requests

```
DELETE /apis/sessions/:sessionId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

403	`{"code":"040006",`
	`"name":"SessionHidden",`
	` "message":"Session hidden",`
	`"httpStatusCode":403}`
		Session is hidden and requester does not own the session.

404	`{"code":"040006",`
	`"name":"SessionNotFound",`
	` "message":"Session not found",`
	`"httpStatusCode":404}`
		Session with UUID `sessionId` not found.

409	`{"code":"060003",`
	`"name":"RecordingInProgress",`
	` "message":"Recording currently in progress",`
	` "httpStatusCode": 409}`
		The session has a recording in progress. All recordings for this session must be stopped before it can be deleted.

Get a List of Videos from a Session

Gets a list of available videos created from the session.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/sessions/sessionId/assets
```

Response

Success

```
{
  "data": [
    {
      "title": "Creator OWN",
      "description": "",
      "ctime": 1395428150,
      "mtime": 1395933239,
      "duration": 6.25,
      "active": false,
      "id": "40de80af-6acb-413c-9c88-04a37b9b1788",
      "movieTrackCount": 1,
      "recording": false,
      "trimming": false,
      "importing": false
    },
    {
      "title": "Creator",
      "description": "",
      "ctime": 1398711729,
      "mtime": 1398711730,
      "duration": 3.45,
      "active": false,
      "id": "b680d35e-f2d5-42e1-9c5e-d12f3fe5dba6",
      "movieTrackCount": 1,
      "recording": false,
      "trimming": false,
      "importing": false
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Errors

Code	Payload	Description
400	`{"code":"010001", "name":"InputValidation", "message":"<message>", "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	The specified session has not generated any recordings.

Get a List of Sessions

Gets a list of sessions for HMP.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/session
```

Tip

You can query sessions based on a title and description (or HotMark title and description) by adding the “ `q` ” query parameter to the collection URL. For more information, see **Queries**.

Response

Success

```
{
  "data": [
    {
      "title": "ABC",
      "description": "",
      "ctime": 1393533035,
      "mtime": 1398875839,
      "duration": 0,
      "active": false,
      "syncrecord": true,
      "id": "2f8dce6d-fd8b-459e-890f-97965c88df40",
      "movieTrackCount": 1,
      "sources": [
        "f4327f7b-3917-4533-b844-8818741cb0bd"
      "recording": false,
      ]
    },
{
      "title": "Creator",
      "description": "",
      "ctime": 1395420810,
      "mtime": 1398190876,
      "duration": 0,
      "active": false,
      "syncrecord": false,
      "id": "f443465b-dee0-484e-b4c3-60fd8209d7b6",
      "movieTrackCount": 5,
      "sources": [
        "8480b370-c6e7-4224-912c-75d9c55bca6c",
        "2d45d18f-2829-4a3d-9bcf-17d74b572467",
        "ac6cf84a-fd2a-4fd1-a6af-feaa336c8857",
        "99c43dce-3b27-4cbf-8a37-6db96e71eb25",
        "323e1db5-5bb4-4988-85fc-b07605b365c8"
      ],
      "recording": false
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
400	{"code":"010002", "name":"InvalidFilter", "message":<Error message>, "httpStatusCode": 400}	The provided filter does not exist on the server.
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	No results returned from the server.
500	{"code":"070000", "name":"InternalServer", "message":"Internal Server Error", "httpStatusCode":500}	Getting a list of sessions failed. Please see `calypso.log` for details.

Session Metadata

Topics include:

- Get a List of Session Metadata
- Edit Session Metadata
- Delete Session Metadata

Get a List of Session Metadata

Gets a list of metadata for the specified session.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/metadata
```

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "creator"
        }
      ]
    },
    {
      "name": "calypso:recording",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "0a354588-b4d1-4f72-acb1- bbcdd3fa75ff"
        }
      ]
    },
    {
      "name": "calypso:creatorDisplayName",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    }
  ]
}
```

Parameters

Property	JSON Type	Property Type	Description
name	String	Required	Name of metadata.

field	Array of objects (fields) for these possible structures:
	`{ "fieldType": "STRING", "valueString": "value" }`
	`{ "fieldType": "INT32", "valueInt32": -14 }`
	`{ "fieldType": "UINT32", "valueUint32": 42 }`
		Required	Content of metadata. See **Fields**.

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode": 404}`	The specified session has no metadata.
404	`{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}`	Session with UUID `sessionId` not found.

Edit Session Metadata

> **⚠ Caution**
>
> Partial updates are *not* supported. The full metadata set submitted in the request will replace the full previous metadata set.

Authorizations: Sessions - EDIT

Requests

```
PUT /apis/sessions/:sessionId/metadata

[
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin7"
        }
      ]
    },
    {
      "name": "calypso:category:Locations",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "Montreal"
        }
      ]
    },
    {
      "name": "calypso:recording",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "6d736ffb-50a2-4068-ad2e- cc147258b3f5"
        }
      ]
    },
    {
      "name": "calypso:creatorDisplayName",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    }
  ]
```

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin7"
        }
      ]
    },
    {
      "name": "calypso:category:Locations",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "Montreal"
        }
      ]
    },
    {
      "name": "calypso:recording",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "6d736ffb-50a2-4068-ad2e- cc147258b3f5"
        }
      ]
    }
  ]
}
```

Error

Code	Payload	Description
400	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
404	`{"code":"040006", "name":"SessionNotFound", "message":"Session not found","httpStatusCode":404}`	Session with UUID `sessionId` not found.

Delete Session Metadata

Deletes the metadata for the session.

Authorizations: Sessions - EDIT

Requests

```
DELETE /apis/sessions/:sessionId/metadata
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"SessionNotFound", "message":"Session  not found", "httpStatusCode":404}`	Session with UUID `sessionId` not found.

Session Share Permissions

Topics include:

- Get Share Permissions for a Session
- Create Share Permissions for a Session
- Get Share Permission Information for a Session
- Edit Share Permission for a Session
- Delete Share Permission for a Session

Get Share Permissions for a Session

The `share` resource is used to manage sharing permissions for the session.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/shares
```

Response

Success

```
{
  "data": [
    {
      "id": "bbc1cbf3-fbee-48dd-8639-920487b6186f",
      "shareType": "USER",
      "permission": "OWN",
      "name": "creator",
      "email": null
    },
    {
      "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
      "shareType": "USER",
      "permission": "WATCH",
      "name": "viewer",
      "email": null
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Parameters

Property	JSON Type	Property Type	Description
id	UUID	Immutable	Group ID or user ID. Identifies the share entity.
shareType	String	Required Immutable	Accepted values, GROUP or USER: • GROUP: the share grants access to the members of a group. • USER: the share grants access to a specific user.
permission	String	Required	Permission level granted to the share's group or user. Accepted values: WATCH, EDIT, OWN, or <uuid> of admin-defined share permission.
name	String	Optional	Group name or user name.

Note

Either id or name must be present. If both are provided, id is used. See User Resources and Group Resources.

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	The specified session has not been shared with any user or group.

Create Share Permissions for a Session

Creates a share permission for a session (i.e., shares the session with the specified user or group).

Authorizations: Sessions - SHARE

Requests

```
POST /apis/sessions/:sessionId/shares

{
    "id": "bbc1cbf3-fbee-48dd-8639-920487b6186f",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "InfoDev",
    "email": null
}
```

Response

Success

```
{
  "data": {
    "id": "bbc1cbf3-fbee-48dd-8639-920487b6186f",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "creator",
    "email": null
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
403	{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}	User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.
404	{"code":"040002", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}	Session with UUID `sessionId` not found.
404	{"code":"040004", "name":"UserNotFound", "message":"User not found", "httpStatusCode":404}	The user with provided `id` or `username` was not found

Code	Payload	Description
404	{"code":"040005", "name":"GroupNotFound", "message":"Group not found", "httpStatusCode":404}	The group with provided `id` or `groupname` was not found.
404	{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The specified share permission was not found.
409	{"code":"060001", "name":"UserHasNoRoles", "message":"User has no roles assigned", "httpStatusCode": 409}	Make sure a role is assigned to the user or to at least one of the user's groups.
409	{"code":"060002", "name":"GroupHasNoRoles", "message":"Group has no roles assigned", "httpStatusCode":409}	Make sure a role is assigned to the group.
500	{"code":"070008", "name":"FindUser", "message":"Error searching for a user", "httpStatusCode":500}	An internal error occurred while searching for user
500	{"code":"070009", "name":"FindGroup", "message":"Error searching for a group", "httpStatusCode":500}	An internal error occurred while searching for group.

Get Share Permission Information for a Session

Gets information about the specified share permission for the session.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/shares/:shareId
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "EDIT",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
404	{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	No user or group was found.

Edit Share Permission for a Session

Edits the specified share permission for the session.

Authorizations: Sessions - SHARE

Requests

```
PUT /apis/sessions/:sessionId/shares/:shareId

{
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "viewer",
    "email": null
}
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

403	`{"code":"030003",`
	`"name":"CannotShare",`
	`"message":"Cannot share",`
	`"httpStatusCode":403}`
		User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.

404	`{"code":"040021",`
	`"name":"UserGroupNotFound",`
	`"message":"User or Group not found",`
	`"httpStatusCode":404}`
		No user or group was found.

404	`{"code": "040025",`
	`"name":"SharePermissionNotFound",`
	`"message": "Share permission not found",`
	`"httpStatusCode": 404}`
		The specified share permission was not found.

Delete Share Permission for a Session

Deletes the specified share permission for the session.

Authorizations: Sessions - SHARE

Requests

```
DELETE /apis/sessions/:sessionId/shares/:shareId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

404	`{"code":"040021",`
	`"name":"UserGroupNotFound",`
	`"message":"User or Group not found",`
	`"httpStatusCode":404}`
		No user or group was found.

Session Sources

Topics include:

- List Session Sources

- Add a Session Source
- Delete Session Sources

List Session Sources

Gets a list of sources assigned to the session. **Authorizations:** Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/sources
```

Response

Success

```
{
  "data": [
    {
      "index": 0,
      "sourceId": "8480b370-c6e7-4224-912c-75d9c55bca6c"
    },
    {
      "index": 1,
      "sourceId": "2d45d18f-2829-4a3d-9bcf-17d74b572467"
    },
    {
      "index": 2,
      "sourceId": "ac6cf84a-fd2a-4fd1-a6af-feaa336c8857"
    },
    {
      "index": 3,
      "sourceId": "323e1db5-5bb4-4988-85fc-b07605b365c8"
    }
  ],
  "paging": {
    "results": 4,
    "pageSize": 15
  }
}
```

Parameters

Property	JSON Type	Property Type	Description
sourceId	string	Required	ID of the source you wish to add to the session. See Source Resources.

Error

Code	Payload	Description
403	`{"code":"040006", "name":"SessionHidden", "message":"Session hidden", "httpStatusCode":403}`	Session is hidden and requester does not own the session.

Code	Payload	Description
404	{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}	Session with UUID `sessionId` not found.
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	Session has no sources.

Add a Session Source

Adds a source to a session. **Authorizations:** Sessions - CHANGE_SOURCES

Requests

```
POST /apis/sessions/:sessionId/sources

{
  "sourceId": "f4327f7b-3917-4533-b844-8818741cb0bd"
}
```

Response

Success

```
{
  "data": {
    "sourceId": "f4327f7b-3917-4533-b844-8818741cb0bd",
    "index": 4
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
403	{"code":"040006", "name":"SessionHidden", "message": "Session hidden", "httpStatusCode": 403}	Session is hidden and requester does not own the session.
404	{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}	Could not find session with the provided id.

Code	Payload	Description
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Source with UUID `sourceId` not found.
409	{"code":"060003", "name":"RecordingInProgress", "message":"Recording currently in progress", "httpStatusCode":409}	There is an active recording on this session.
409	{"code":"060008", "name":"SessionSourceAlreadyExists", "message":"Source already added to this session", "httpStatusCode":409}	This source was already added to this session.

Delete Session Sources

Deletes the specified source from the session.

Authorizations: Sessions - CHANGE_SOURCES

Requests

```
DELETE /apis/sessions/:sessionId/sources/:sourceId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
403	{"code":"040006", "name":"SessionHidden", "message": "Session hidden", "httpStatusCode": 403}	Session is hidden and requester does not own the session.
404	{"code":"040006", "name":"SessionNotFound", "message": "Session not found", "httpStatusCode": 404}	Session with UUID `sessionId` not found.
404	{"code":"040009", "name":"SourceNotFound", "message":"Source not found", "httpStatusCode":404}	Source with UUID `sourceId` not found.

Code	Payload	Description
404	{"code":"040015","name":"TrackNotFound" , "message":"Track not found", "httpStatusCode":404}	The track could not be found.
409	{"code":"060003", "name":"RecordingInProgress", "message":"Recording currently in progress", "httpStatusCode":409}	There is an active recording on this session.

Recording Sessions

Topics include:

- Get a List of Active Recordings for a Session
- Create a New Recording Session

Get a List of Active Recordings for a Session

Gets a list of currently active recordings for the session.

Authorizations: Sessions - RECORD

Requests

```
GET /apis/sessions/:sessionId/recordings
```

Response

Success

```
{
  "data": [
    {
      "id": "d0408eaa-a512-4256-813f-96c0b3f47bd0",
      "session": "f443465b-dee0-484e-b4c3-60fd8209d7b6",
      "username": "haiadmin",
      "state": "RECORDING",
      "duration": 25.638,
      "recorders": [
        {
          "source": "323e1db5-5bb4-4988-85fc-b07605b365c8",
          "chunk": "asset-chunk://d0408eaa-a512-4256-813f-96c0b3f47bd0/3/0/0",
          "id": "6",
          "state": "RECORDING"
        }
      ],
      "ctime": 1399402432,
      "title": "Creator"
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
`404`	`{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}`	Session with UUID `sessionId` not found.
`404`	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	No results returned from the server.

Create a New Recording Session

Creates a new active recording of the session. Certain metadata, like the title and description, are automatically copied from the session. The sources to record can be specified in the request. By default, the active recording records all of the session’s sources. Sources that are not associated with the session are ignored.

Authorizations: Sessions - RECORD

Requests

```
POST /apis/sessions/:sessionId/recordings

{
  "recorders": [
  {
    "source": "323e1db5-5bb4-4988-85fc-b07605b365c8"
  },
  {
    "source": "54fde7cd-c734-470e-b188-5d60def4f63a"
  }
  ]
}
```

Response

Success

```
{
  "data": {
    "id": "bea38704-ef95-4330-9f77-ad52a2c5a8b3",
    "session": "ad0eff6f-12b3-4622-893e-a5586046ab82",
    "username": "haiadmin",
    "state": "RECORDING",
    "duration": 0,
    "recorders": [
      {
        "source": "323e1db5-5bb4-4988-85fc-b07605b365c8",
        "chunk": "asset-chunk://bea38704-ef95-4330-9f77-ad52a2c5a8b3/3/0/0",
        "id": "7"
      },
      {
        "source": "54fde7cd-c734-470e-b188-5d60def4f63a",
        "chunk": "asset-chunk://bea38704-ef95-4330-9f77-ad52a2c5a8b3/4/0/0",
        "id": "8"
      }
    ],
    "ctime": 1399407563,
    "title": "InfoDevSession - 06/05/2014, 15:26"
  }
}
```

Error

Code	Payload	Description
`403`	`{"code":"040006", "name":"SessionHidden", "message":"Session hidden", "httpStatusCode":403}`	Session is hidden and requester does not own the session.
`404`	`{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}`	Session with UUID `sessionId` not found.

Code	Payload	Description

409	`{"code":"060009","name":"SessionHasNoSource",`
	`"message":"Session requires at least one source",`
	`"httpStatusCode":409}`
		Session must have at least one source.

500	`{"code":"070027","name":"InsufficientSpace",`
	`"message":"Insufficient disk space on server to`
	`complete operation",`
	`"httpStatusCode":500}`
		The server has run out of disk space.

Get Thumbnail Image for a Session

Retrieves the thumbnail image for the specified session. Responds with the thumbnail for the first track.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/thumbnail
```

Response

Success

```
Thumbnail in JPEG format.
```

Error

Code	Payload	Description

404	`{"code":"040011",`
	`"name":"ThumbnailNotFound",`
	`"message":"Thumbnail not found",`
	`"httpStatusCode":404}`
		The session is missing a thumbnail.

404	`{"code":"040002",`
	`"name":"SessionNotFound",`
	`"message":"Session not found",`
	`"httpStatusCode":404}`
		Could not find session with the provided id.

Public Share Actions

Topics include:

- Get a List of Public Shares
- Create a Public Share

- Delete Public Share

Get a List of Public Shares

Gets a list of public shares created for the specified session.

Authorizations: Sessions - SHARE

Requests

```
GET /apis/sessions/:sessionId/publicshare
```

Response

Success

```
{
 "data":{"permission":"WATCH"}
}
```

Error

Code	Payload	Description
404	{"code": "040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The specified share permission was not found.

Create a Public Share

Note

Only the WATCH permission should be used. For example: {“permission”: “WATCH”}

Authorizations: Sessions - WATCH

Requests

```
POST /apis/sessions/:sessionId/publicshare

{
  id: "fc2b42e6-cf75-43f0-822e-e8b25edba217",
  resource: "sessions",
  permission: "WATCH"
}
```

Response

Success

```
{
  "data":{
    "permission":"WATCH"
  }
}
```

Error

Code	Payload	Description
400	`{"code": "010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
401	`{"code":"020001", "name":"UserNotAuthorized", "message":"User is not authorized", "httpStatusCode":401}`	User is not authenticated.
403	`{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}`	User is not allowed to share to this permission ID.
404	`{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}`	The specified share permission was not found.

Delete Public Share

Deletes the specified public share for the session. **Authorizations:** Sessions - SHARE

Requests

```
DELETE /apis/sessions/:sessionId/publicshare
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
400	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.

Code	Payload	Description
403	`{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}`	User is not allowed to share to this permission ID.
404	`{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}`	The specified share permission was not found.
404	`{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}`	The specified user or group was not found.
404	`{"code": "040004", "name":"UserNotFound", "message": "User not found", "httpStatusCode":404}`	The user with provided `id` or `username` was not found.
404	`{"code":"040006", "name":"SessionNotFound", "message":"Session not found", "httpStatusCode":404}`	Unable to locate session.

Event Actions

Topics include:

- Get an Event List
- Create a Session Event
- Get Event Information for a Session
- Edit a Session Event

Get an Event List

Gets a list of events associated with the session. For a list of event fields, types and descriptions, see Event Resources.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/events
```

Response

Success

```
{
  "data": [
    {
      "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
      "title": "Session - 10/21/2015, 2:04 PM",
      "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
      "startTime": 1445452200,
      "duration": 1800,
      "type": "weekly-recurring",
      "typeData": {
        "days": ["MON", "WED", "SAT"]
      },
      "command": "set-live",
      "parameters": {
        "record": true
      }
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
`400`	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
`404`	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	The specified session has not generated any events.

Create a Session Event

Creates an event associated with a session. **Authorizations:** Sessions - EDIT

Requests

```
POST /apis/sessions/:sessionId/events

{
  "title": "InfoDevSource",
  "description": "UDP",
  "syncrecord": true
}
```

Response

Success

```
{
  "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
  "title": "Session - 10/21/2015, 2:04 PM",
  "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
  "startTime": 1445452200,
  "duration": 1800,
  "type": "weekly-recurring",
  "typeData": {
    "days": ["MON", "WED", "SAT"]
  },
  "command": "set-live",
  "parameters": {
    "record": true
  }
}
```

Error

HTTP Status Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

500	`{"code":"070019",`
	`"name":"CreateSession",`
	`"message":"Failed to create a session",`
	`"httpStatusCode":500}`
		Creating the session failed. Please see `calypso.log` for details.

Get Event Information for a Session

Gets information for an event associated with the session. For a list of event fields, types and descriptions, see Event Resources.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/sessions/:sessionId/events/:eventId
```

Response

Success

```
{
  "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
  "title": "Session - 10/21/2015, 2:04 PM",
  "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
  "startTime": 1445452200,
  "duration": 1800,
  "type": "weekly-recurring",
  "typeData": {
    "days": ["MON", "WED", "SAT"]
  },
  "command": "set-live",
  "parameters": {
    "record": true
  }
}
```

Error

Code	Payload	Description

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

404	`{"code": "040012", "name":"NoResults",`
	`"message":"No results",`
	`"httpStatusCode":404}`
		The specified session has not generated any events.

Edit a Session Event

Edits an event associated with the session. **Authorizations:** Sessions - EDIT

Requests

```
PUT /apis/sessions/:sessionId/events/:eventId

{
  "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
  "title": "Session - 10/21/2015, 2:04 PM",
  "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
  "startTime": 1445452200,
  "duration": 1800,
  "type": "weekly-recurring",
  "typeData": {
    "days": ["MON", "WED", "SAT"]
  },
  "command": "set-live",
  "parameters": {
    "record": true
  }
}
```

Response

Success

```
{
  "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
  "title": "Session - 10/21/2015, 2:04 PM",
  "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
  "startTime": 1445452200,
  "duration": 1800,
  "type": "weekly-recurring",
  "typeData": {
    "days": ["MON", "WED", "SAT"]
  },
  "command": "set-live",
  "parameters": {
    "record": true
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
403	{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}	User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.
404	{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	No user or group was found.
404	{"code": "040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode": 404}	The specified share permission was not found.

Recording Resources

Synopsis

An *active* recording is a recording that is currently being created by recording a session's sources. When it is paused, it becomes a paused recording. When it is stopped, it becomes a finished recording. Finished recordings are referred to as *videos*.

The recordings API is used to list, pause/resume, delete, and insert HotMarks in active recordings.

Recording Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the resource. Generated by HMP.
session	String	Immutable	The session being recorded.
username	String	Immutable	The name of the user making the recording.
state	String	Optional	The desired state of the recording. Accepted values: "PAUSED" or "RECORDING".
duration	Number	Ignored	The length of the recording in seconds.
recorders	Array	Optional	Contains source IDs on which to start recording. If parameter is not provided, recording will be started on all of session's sources.
ctime	Number	Ignored	The creation time of the recording as a Unix epoch timestamp (seconds).
title	String	Immutable	Title of the recording.

Get a List of Active Recordings

Gets a list of currently active recordings for HMP.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/recordings
```

Response

Success

```
{
 "data": [
   {
     "id": "6d736ffb-50a2-4068-ad2e-cc147258b3f5",
     "session": "ad0eff6f-12b3-4622-893e-a5586046ab82",
     "username": "haiadmin",
     "state": "RECORDING",
     "duration": 42.155,
     "recorders": [
       {
         "source": "54fde7cd-c734-470e-b188-5d60def4f63a",
         "chunk": "asset-chunk://6d736ffb-50a2-4068-ad2e-cc147258b3f5/3/0/0",
         "id": "14",
         "state": "RECORDING"
       },
       {
         "source": "f4327f7b-3917-4533-b844-8818741cb0bd",
         "chunk": "asset-chunk://6d736ffb-50a2-4068-ad2e-cc147258b3f5/4/0/0",
         "id": "15",
         "state": "RECORDING"
       }
     ],
     "ctime": 1399586996,
     "title": "InfoDevSession - 06/05/2014, 15:26"
   },
   {
     "id": "fa20c493-7d40-4c8d-919c-8e370482682c",
     "session": "f443465b-dee0-484e-b4c3-60fd8209d7b6",
     "username": "haiadmin",

      "state": "RECORDING",
   "duration": 18.816,
    "recorders": [
        {
         "source": "323e1db5-5bb4-4988-85fc-b07605b365c8",
 "chunk": "asset-chunk://fa20c493-7d40-4c8d-919c-8e370482682c/3/0/0",
  "id": "16",
         "state": "RECORDING"
        }
     ],
     "ctime": 1399587019,
 "title": "Creator"
 }
 ],
 "paging": {
   "results": 2,
   "pageSize": 15
 }
}
```

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	There are no active recordings.

Get Recording Details

Gets details on an active recording.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/recordings/:recordingId
```

Response

Success

```
{
  "data": {
    "id": "bea38704-ef95-4330-9f77-ad52a2c5a8b3",
    "session": "ad0eff6f-12b3-4622-893e-a5586046ab82",
    "username": "haiadmin",
    "state": "RECORDING",
    "duration": 405.03,
    "recorders": [
      {
        "source": "323e1db5-5bb4-4988-85fc-b07605b365c8",
        "chunk": " asset-chunk://bea38704-ef95-4330-9f77- ad52a2c5a8b3/3/0/0",
        "id": "7"
      },
      {
        "source": "54fde7cd-c734-470e-b188-5d60def4f63a",
        "chunk": " asset-chunk://bea38704-ef95-4330-9f77- ad52a2c5a8b3/4/0/0",
        "id": "8"
      }
    ],
    "ctime": 1399407563,
    "title": "InfoDevSession - 06/05/2014, 15:26"
  }
}
```

Error

Code	Payload	Description
404	{"code":"040001", "name":"RecordingNotFound", "message":"Active recording not found", "httpStatusCode":404}	Recording with UUID `recordingId` not found.

Pause or Resume a Recording

Pauses or resumes an active recording. PUT method accepts an object with a state property that indicates the desired state of the recording: PAUSED or RECORDING.

Authorizations: Recordings - RECORD

Requests

```
PUT /apis/recordings/:recordingId

{
  "state": "PAUSED"
}
```

Response

Success

```
{
  "data": {
    "id": "bea38704-ef95-4330-9f77-ad52a2c5a8b3",
    "session": "ad0eff6f-12b3-4622-893e-a5586046ab82",
    "username": "haiadmin",
    "state": "PAUSED",
    "duration": 1196.283,
    "recorders": [
      {
        "source": "323e1db5-5bb4-4988-85fc-b07605b365c8",
        "chunk": "asset-chunk://bea38704-ef95-4330-9f77-ad52a2c5a8b3/3/0/0",
        "id": "7"
      },
      {
        "source": "54fde7cd-c734-470e-b188-5d60def4f63a",
        "chunk": "asset-chunk://bea38704-ef95-4330-9f77-ad52a2c5a8b3/4/0/0",
        "id": "8"
      }
    ],
    "ctime": 1399407563,
    "title": "InfoDevSession - 06/05/2014, 15:26"
  }
}
```

Error

Code	Payload	Description

404	`{"code":"040001",`
	`"name":"RecordingNotFound",`
	`"message":"Active recording not found",`
	`"httpStatusCode":404}`
		Recording with UUID `recordingId` not found.

400	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":"State should be either'RECORDING' or PAUSED'",`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

Stop a Recording

Stops the recording. It is no longer accessible as a recording and must be accessed as a video.

Authorizations: Recordings - RECORD

Requests

```
DELETE /apis/recordings/:recordingId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

404	`{"code":"040001",`
	`"name":"RecordingNotFound",`
	`"message":"Active recording not found",`
	`"httpStatusCode":404}`
		Recording with UUID `recordingId` not found.

Create a HotMark on a Recording

Creates a new HotMark on the specified recording.

Authorizations: Recordings - EDIT

Requests

```
POST /apis/recordings/:recordingId/hotmarks

{
  "timestamp": 2000,
  "title": "HotMark1",
  "description": "Training Session"
}
```

Response

Success

```
{
  "data": {
    "timestamp": 2000,
    "title": "HotMark1",
    "description": "Training Session",
    "id": "ed6397d0-e947-4604-84f2-96b9ca68d72a"
    "username":"jsmith",
    "userDisplayName":"John Smith"
  }
}
```

Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the HotMark. Generated by HMP.
timestamp	Number	Required for videos. Optional (for POST) for active recordings.	**Note** HotMark timestamps have a one-second resolution.
title	String	Required	Title of the HotMark.
description	String	Optional	Description of the HotMark.
username	String	Immutable	The user's login name, e.g., bsmith
userDisplayName	String	Ignored	The user’s friendly name, e.g., Bob Smith

Error

Code	Payload	Description
404	`{"code":"040001","name":"RecordingNotFound", "message":"Active recording not found", "httpStatusCode":404}`	Recording with UUID `recordingId` not found.
400	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
409	`{"code":"060004", "name":"HotmarkAlreadyExists", "message": "HotMark already exists at this time", "httpStatusCode": 409}`	HotMark already exists at this timestamp.
404	`{"code":"010004", "name":"NoRecordingActive", "message":"'Y' is not an active recording", "httpStatusCode": 404}`	Active recording not found.

Category Resources

Synopsis

> **Note**
>
> Category setup and administration must be done from the Administration Web interface.

The `categories` API may be used to get a list of categories. Category values are stored in sessions and videos using their `metadata` APIs.

Category Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the category, generated by HMP.
label	String	Required	The name used to identify the category. Category labels must be unique.
values	Array of strings	Optional	**Note** This list is not updated with user-defined dynamic values.
multipleValues	Boolean	Required	`false` : Only one value can be set on a video for that category. `true` : One or more values can be assigned.
allowCustomValues	Boolean	Required	`false` : Only predefined values may be assigned. `true` : Predefined or custom values may be assigned.
customValues	Array of strings	Optional	**Note** Contains the userdefined “dynamic” values.
index	number	Optional	Order in which categories are returned and displayed.
group	string	Optional	Used to assign a group to a category. The categories are grouped when displayed.

Get a List of Categories

Gets a list of all categories defined on the HMP system.

Authorizations: Categories - VIEW

Requests

```
GET /apis/categories
```

Response

Success

```
{
  "data": [
    {
      "label": "Locations",
      "values": ["Montreal", "Chicago", "Austin"],
      "multipleValues": true,
      "allowCustomValues": true,
      "id": "5ee67847-d886-4fda-b755-ce4090fa8739",
      "customValues": [
        "Minneapolis"
      ]
      "index": 1,
      "group": "Teams"
    },
    {
      "label": "Sports",
      "values": ["Hockey", "Football", "Tennis"],
      "multipleValues": true,
      "allowCustomValues": true,
      "id": "102d21c2-3564-40cd-aed8-7e57ec6c9061",
      "customValues": []
      "index": 2,
      "group": "Teams"
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	No categories are defined.

Get Category Details

Gets details about the specified category.

Authorizations: Categories - VIEW

Requests

```
GET /apis/categories/:id
```

Response

Success

```
{
  "data": {
    "label": "Locations",
    "values": ["Montreal", "Chicago", "Austin"],
    "multipleValues": true,
    "allowCustomValues": true,
    "id": "5ee67847-d886-4fda-b755-ce4090fa8739",
    "customValues": [
      "Minneapolis"
    ]
    "index": 1,
    "group": "Teams"
  }
}
```

Error

Code	Payload	Description

404	`{"code":"040007",`
	`"name":"CategoryNotFound",`
	`"message":"Category not found",`
	`"httpStatusCode":404}`
		A category with a UUID of `id` was not found.

Asset Resources

Synopsis

The `assets` API is used to manage videos.

Asset Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the resource, generated by HMP.
title	String	Required	Title of the video.
description	String	Optional	Description of the video.
ctime	Number	Ignored	The creation time of the video as a Unix epoch timestamp (seconds).
mtime	Number	Ignored	The last modification time of the video as a Unix epoch timestamp (seconds).
duration	Number	Ignored	The length of the video in seconds
active	Boolean	—	Always false for videos and is unused.
movieTrackCount	Number	Ignored	The number of movie tracks in the video. This represents how many sources were part of the recording. When recording a session with 4 sources, the video has 4 movie tracks.
recording	Boolean	Ignored	Read-only state information. True when the video is currently being recorded; false when the recording is complete.

Property	JSON Type	Property Type	Description
trimming	Boolean	Ignored	Read-only state information. True when the video is currently being trimmed from another video; false when trimming is complete.
importing	Boolean	Ignored	Read-only state information. True when the video is currently being imported to the system; false when import is complete.

Video Actions

Topics include:

- Get a List of Recorded Videos
- Get Video Details
- Update a Title or Description
- Delete a Video
- Get List of Related Videos
- Download Videos

Get a List of Recorded Videos

Lists the videos stored on the HMP system. **Authorizations:** Recordings - VIEW

Requests

```
GET /apis/assets
```

✓ **Tip**

You can query videos based on a title and description (or HotMark title and description) by adding the “ `q` ” query parameter to the collection URL. For more information, see Queries.

Response

Success

```
{
  "data": [
    {
      "id": "3bc946a6-6720-41c0-8246-55e8119d1a5a",
      "title": "New Session",
      "description": "",
      "active": false,
      "ctime": 1401997809,
      "mtime": 1401997809,
      "tag": null,
      "duration": 60.347000000000001,
      "recording": false,
      "trimming": false,
      "importing": false,
      "movieTrackCount": 1
    },
    {
      "title": "New Session",
      "description": "",
      "ctime": 1402000846,
      "mtime": 1402000846,
      "duration": 0,
      "active": false,
      "id": "19db79ba-bbde-4c73-9543-8c169d7b1383",
      "movieTrackCount": 2,
      "recording": true,
      "trimming": false,
      "importing": false
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
400	{"code":"010002", "name":"InvalidFilter", "message":<Error message>, "httpStatusCode":400}	The provided filter could not be parsed.
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	There are no videos, or no videos matched the specified search query and/or filter query.

Get Video Details

Gets details on the specified video.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id
```

Response

Success

```
{
  "data": {
    "id": "3bc946a6-6720-41c0-8246-55e8119d1a5a",
    "title": "New Session",
    "description": "",
    "active": false,
    "ctime": 1401997809,
    "mtime": 1401997809,
    "duration": 60.347000000000001,
    "recording": false,
    "trimming": false,
    "importing": false,
    "movieTrackCount": 1
  }
}
```

Error

Code	Payload	Description
`404`	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.

Update a Title or Description

Updates the title and/or description of the specified video.

Authorizations: Recordings - EDIT

Requests

```
PUT /apis/assets/:id

{
  "title": "InfoDevSession - 18/03/2014, 15:06",
  "description": "API Test"
}
```

Response

Success

```
{
  "data": {
    "title": "InfoDevSession - 18/03/2014, 15:06",
    "description": "API Test",
    "active": false,
    "id": "a6af4f7d-5c2d-4550-ace8-0b6a17c78540",
    "ctime": 1395950679,
    "mtime": 1399488944,
    "duration": 400.8,
    "recording": false,
    "trimming": false,
    "importing": false,
    "movieTrackCount": 1
  }
}
```

Error

Code	Payload	Description

`400`	`{"code":"010001",`
	`"name":"InputValidation",`
	`"message":<Error message>,`
	`"httpStatusCode":400}`
		Supplied data was insufficient or incorrect.

`404`	`{"code":"040002",`
	`"name":"AssetNotFound",`
	`"message":"Video not found",`
	`"httpStatusCode":404}`
		Video with UUID `Id` not found.

`500`	`{"code":"070017", "name":"UpdateAsset",`
	`"message":"Failed to update a video",`
	`"httpStatusCode":500}`
		Unable to write updated data to video.

Delete a Video

Deletes the specified video.

Authorizations: Recordings - DELETE

Requests

```
DELETE /apis/assets/:id
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

404	`{"code":"040002",`
	`"name":"AssetNotFound",`
	`"message": "Video not found",`
	`"httpStatusCode": 404}`
		Video with UUID `Id` not found.

409	`{"code":"060003",`
	`"name":"RecordingInProgress",`
	`"message":"Recording currently in progress",`
	`"httpStatusCode":409}`
		Recordings cannot be deleted while they are actively recording.

409	`{"code":"060005",`
	`"name":"RecordingBeingTrimmmed",`
	`"message":"Video currently being trimmed",`
	`"httpStatusCode":409}`
		Videos cannot be deleted while they are being trimmed.

409	`{"code":"060006",`
	`"name":"RecordingBeingImported",`
	`"message":"Video currently being imported",`
	`"httpStatusCode":409}`
		Videos cannot be deleted while they are being imported.

409	`{"code":"060007",`
	`"name":"RecordingBeingExported",`
	`"message":"Video currently being exported",`
	`"httpStatusCode":409}`
		Videos cannot be deleted while they are being exported.

409	`{"code":"060013",`
	`"name":"RecordingBeingStreamed",`
	`"message":"Video currently being streamed",`
	`"httpStatusCode":409}`
		The video is being streamed.

Get List of Related Videos

Lists related videos based on matching category data.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/related
```

Response

Success

```
{
  "data": [
    {
      "id": "3bc946a6-6720-41c0-8246-55e8119d1a5a",
      "title": "New Session",
      "description": "",
      "active": false,
      "ctime": 1401997809,
      "mtime": 1401997809,
      "tag": null,
      "duration": 60.34700000000001,
      "recording": false,
      "trimming": false,
      "importing": false,
      "movieTrackCount": 1
    },
    {
      "title": "New Session",
      "description": "",
      "ctime": 1402000846,
      "mtime": 1402000846,
      "duration": 0,
      "active": false,
      "id": "19db79ba-bbde-4c73-9543-8c169d7b1383",
      "movieTrackCount": 2,
      "recording": true,
      "trimming": false,
      "importing": false
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Could not find video with the provided id.
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	No results returned from the server.
500	`{"code":"070000", "name":"InternalServer", "message":"Internal Server Error", "httpStatusCode":500}`	Getting a list of related videos failed. Please see `calypso.log` for details.

Download Videos

Downloads a ZIP file containing a the specified videos in TS or MP4 format.

Authorizations: Recordings - DOWNLOAD

Requests

```
GET /apis/assets/:id/download
```

The file type of the downloaded file may be specified with a field included in the download URLs query component.

Field	Required	Validation	Description
`fileType`	no (defaults to ts)	ts or mp4	ts: Downloads videos in Transport Stream (TS) file format. mp4: Downloads videos in MPEG-4 file format.

Example:

```
GET /apis/assets/2e05d5bd-2d1e-477a-8f62-92301362c7fd/download?fileType=ts
```

Response

Success

```
Zipped asset in TS or MP4 format.
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID Id not found.
409	`{"code":"060019", "name":"RecordingBeingProcessed", "message":"Video currently being processed", "httpStatusCode":409}`	Video with UUID Id currently being processed.

Video Metadata

Topics include:

- Get Video Metadata
- Edit Video Metadata
- Delete Video Metadata

Get Video Metadata

Gets the metadata for the specified video. See Get a List of Session Metadata.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/metadata
```

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    },
    {
      "name": "calypso:recorded_from_session",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "e4e7c695-5d07-4646-b849- ee79a11d92aa"
        }
      ]
    },
    {
      "name": "calypso:creatorDisplayName",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    }
  ]
}
```

Error

HTTP Status Code	Payload	HTTP Status Code

`404`	`{"code":"040012",`
	`"name":"NoResults",`
	`"message":"No results",`
	`"httpStatusCode":404}`
		Specified video contains no metadata.

`404`	`{"code":"040002",`
	`"name":"AssetNotFound",`
	`"message":"Video not found",`
	`"httpStatusCode":404}`
		Video with UUID `Id` not found.

Edit Video Metadata

> **Caution**
>
> Partial updates are not supported. The full metadata set submitted in the request will replace the full previous metadata set.

Authorizations: Recordings - EDIT

Requests

```
PUT /apis/assets/:id/metadata

[
  {
  "name": "calypso:creator",
  "field": [
    {
    "fieldType": "STRING",
    "valueString": "haiadmin"
    }
  ]
  },
  {
  "name": "calypso:recorded_from_session",
  "field": [
    {
    "fieldType": "STRING",
    "valueString": "e4e7c695-5d07-4646-b849-ee79a11d92aa"
    }
  ]
  },
  {
  "name": "calypso:category:Locations",
  "field": [
    {
    "fieldType": "STRING",
    "valueString": "Chicago"
    },
    {
    "fieldType": "STRING",
    "valueString": "Montreal"
    }
  ]
  }
]
```

Response

Success

```
{
  "data": [
    {
      "name": "calypso:creator",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    },
    {
      "name": "calypso:recorded_from_session",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "e4e7c695-5d07-4646-b849- ee79a11d92aa"
        }
      ]
    },
    {
      "name": "calypso:category:Locations",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "Chicago"
        },
        {
          "fieldType": "STRING",
          "valueString": "Montreal"
        }
      ]
    },
    {
      "name": "calypso:creatorDisplayName",
      "field": [
        {
          "fieldType": "STRING",
          "valueString": "haiadmin"
        }
      ]
    }
  ]
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
404	{"code":"040006", "name":"AssetNotFound", "message":"Video not found","httpStatusCode":404}	Video with UUID `Id` not found.

Delete Video Metadata

> ⚠ **Caution**
>
> The full metadata set is cleared.

Authorizations: Recordings - EDIT

Requests

```
DELETE /apis/assets/:id/metadata
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.

Video Share Permissions

Topics include:

- Get Video Sharing Permission Information
- Create Video Share Permissions
- Get Video Share Permissions
- Edit Video Share Permissions
- Delete Video Share Permissions

Get Video Sharing Permission Information

The `share` resource is used to manage sharing permissions for a video. See Get Share Permissions for a Session.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/shares
```

Response

Success

```
{
  "data": [
    {
      "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
      "shareType": "USER",
      "permission": "WATCH",
      "name": "viewer",
      "email": null
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	The specified video has not been shared with any user or group.

Create Video Share Permissions

Creates a share permission on the specified video. **Authorizations:** Recordings - SHARE

Requests

```
POST /apis/assets/:id/shares

{
 "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
 "shareType": "USER",
 "permission": "EDIT",
 "name": "viewer"
}
```

Note

Either `id` or `name` must be present. If both are provided, `id` will be used. See User Resources and Group Resources.

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "EDIT",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
403	{"code": "030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}	User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.
404	{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}	Video with UUID `Id` not found.
404	{"code":"040004", "name":"UserNotFound", "message":"User not found", "httpStatusCode":404}	User with provided `id` or `name` was not found.
404	{"code":"040005", "name":"GroupNotFound", "message":"Group not found", "httpStatusCode":404}	Group with provided `id` or `name` was not found.
404	{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The share permission to share with was not found.
409	{"code":"060001", "name":"UserHasNoRoles", "message":"User has no roles assigned", "httpStatusCode":409}	Make sure a role is assigned to the user or to at least one of the user's groups.
409	{"code":"060002", "name":"GroupHasNoRoles", "message":"Group has no roles assigned", "httpStatusCode":409}	Make sure a role is assigned to the group.

Get Video Share Permissions

Gets information about the specified share for a video. See Get Share Permissions for a Session.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/shares/:shareId
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "EDIT",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
403	{"code":"030003", "name":"CannotShare", "message": "Cannot share", "httpStatusCode": 403}	User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.
404	{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	A share for the specified user or group ID does not exist.
404	{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The share permission to share with was not found.

Edit Video Share Permissions

Edits the specified share for a video.

Authorizations: Recordings - SHARE

Requests

```
PUT /apis/assets/:id/shares/:shareId

{
  "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
  "shareType": "USER",
  "permission": "WATCH",
  "name": "viewer"
}
```

Response

Success

```
{
  "data": {
    "id": "a78758cc-48da-47ad-bc6e-6c8564badb7e",
    "shareType": "USER",
    "permission": "WATCH",
    "name": "viewer",
    "email": null
  }
}
```

Error

Code	Payload	Description
403	`{"code":"030003", "name":"CannotShare", "message":"Cannotshare", "httpStatusCode":403}`	User is not allowed to share to this permission ID. Ex.: User has the 'WATCH' share permission but is trying to share to the 'OWN' level.
404	`{"code":"040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}`	Specified video has no share for the specified user or group ID.
400	`{"code":"010001", "name":"InputValidation", "message":, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
404	`{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}`	The share permission to share with was not found.

Delete Video Share Permissions

Deletes the specified share for a video.

Authorizations: Recordings - SHARE

Requests

```
DELETE /apis/:id/shares/:shareId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

404	`{"code":"040021",`
	`"name":"UserGroupNotFound",`
	`"message":"User or Group not found",`
	`"httpStatusCode":404}`
		Specified video has no share for the specified user or group ID.

Video HotMarks

Topics include:

- Get a List of HotMarks
- Create a New HotMark
- Get Specific HotMark Information
- Update a Hotmark
- Delete a HotMark

Get a List of HotMarks

Gets HotMark information for the specified video. See Create a HotMark on a Recording.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/hotmarks
```

Response

Success

```
{
  "data": [
    {
      "title": "Hotmark POST Test",
      "description": "API added HM",
      "timestamp": 100000,
      "id": "0d75aa67-3866-4e74-a285-011e951ca0af"
      "username":"jsmith",
      "userDisplayName":"John Smith"
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040002","name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.
404	`{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}`	This video contains no HotMarks.

Create a New HotMark

Creates a new HotMark on the specified video. **Authorizations:** Recordings - EDIT

Requests

```
PUT /apis/assets/:id/hotmarks

{
  "title": "Hotmark POST Test",
  "description": "API added HM",
  "timestamp": 100000
}
```

Response

Success

```
{
  "data": {
    "title": "Hotmark POST Test",
    "description": "API added HM",
    "timestamp": 100000,
    "id": "0d75aa67-3866-4e74-a285-011e951ca0af"
    "username":"jsmith",
    "userDisplayName":"John Smith"
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Video with UUID `Id` not found.
404	{"code":"040002","name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}	The video could not be found.
409	{"code":"060004", "name":"HotmarkAlreadyExists", "message":"HotMark already exists at this time", "httpStatusCode":409}	HotMark already exists at this time.

Get Specific HotMark Information

Gets information about the specified HotMark for a video.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/hotmarks/:hotmarkId
```

Response

Success

```
{
  "data": {
    "title": "Hotmark POST Test",
    "description": "API added HM",
    "timestamp": 100000,
    "id": "0d75aa67-3866-4e74-a285-011e951ca0af"
    "username":"jsmith",
    "userDisplayName":"John Smith"
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.
404	`{"code":"040010", "name":"HotmarkNotFound", "message":"HotMark not found", "httpStatusCode":404}`	HotMark with UUID `hotmarkId` not found.

Update a Hotmark

Updates the specified HotMark.

Authorizations: Recordings - EDIT

Requests

```
PUT /apis/assets/:id/hotmarks/:hotmarkId

{
  "title": "Hotmark PUT Test",
  "description": "API updated HM",
  "timestamp": 120000
}
```

Response

Success

```
{
  "data": {
    "title": "Hotmark PUT Test",
    "description": "API updated HM",
    "timestamp": 120000,
    "id": "0d75aa67-3866-4e74-a285-011e951ca0af"
    "username":"jsmith",
    "userDisplayName":"John Smith"
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.
400	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.

Code	Payload	Description

404	`{"code":"040010",`
	`"name":"HotmarkNotFound",`
	`"message":"HotMark not found",`
	`"httpStatusCode":404}`
		HotMark with UUID `hotmarkId` not found.

409	`{"code":"060004",`
	`"name":"HotmarkAlreadyExists",`
	`"message":"HotMark already exists at`
	`this time",`
	`"httpStatusCode":409}`
		HotMark already exists at this time.

Delete a HotMark

Deletes the specified HotMark.

Authorizations: Recordings - EDIT

Requests

```
DELETE /apis/assets/:id/hotmarks/:hotmarkId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description

404	`{"code":"040002", "name":"AssetNotFound",`
	`"message":"Video not found",`
	`"httpStatusCode":404}`
		Video with UUID `Id` not found.

404	`{"code":"040010", "name":"HotmarkNotFound",`
	`"message":"HotMark not found",`
	`"httpStatusCode":404}`
		HotMark with UUID `hotmarkId` not found.

Stream Assets

Topics include:

- Get a Stream List Started from a Video
- Create a Stream
- Get Stream Information for a Video
- Delete a Stream for a Video

Get a Stream List Started from a Video

Gets a list of streams that play back the specified video on the network.

Authorizations: Recordings - STREAM

Requests

```
GET /apis/assets/:id/streams
```

Response

Success

```
{
  "data": [
    {
      "address": "239.134.32.100",
      "port": 4900,
      "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
      "username": "haiadmin",
      "id": "02f4344f-35cc-455f-bcfd-1b316550a87c"
    }
  ],
  "paging": {
    "results": 1,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.
404	`{"code":"040012","name":"NoResults", "message":"No results", "httpStatusCode":404}`	There are no streams playing back the video.

Create a Stream

Note

For multi-source videos, the “address” and “port” are ignored, and a “destinations” list must be present.

Authorizations: Recordings - STREAM

Requests

Stream:

```
POST /apis/assets/:id/streams

{
  "address": "239.134.32.100",
  "port": 4900
}
```

Multi-source Video:

```
POST /apis/assets/:id/streams

{
   "address": "ignored",
   "port": 1,
   "destinations": [{
     "trackId": 3,
     "address": "239.134.32.100",
     "port": 4900
   }, {
     "trackId": 4,
     "address": "239.134.32.100",
     "port": 4901
   }]
}
```

Response

Success

Stream:

```
{
  "data": {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "5e33bc64-4ff7-47aa-a7d6-562b35a9c4aa"
  }
}
```

Multi-source Video:

```
{
  "data": {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "5e33bc64-4ff7-47aa-a7d6-562b35a9c4aa",
     "destinations": [{
       "trackId": 3,
       "address": "239.134.32.100",
       "port": 4900
     }]
  }
}
```

Note

For multiple destinations, “address” and “port” are overwritten respectively to “multiple” and 0.

Error

Code	Payload	Description
`400`	`{"code":"010005", "name":"MultiSourcesStream", "message":"Multi-Source recordings must have a destination list", "httpStatusCode":400}`	You must provide a list of destinations for videos that contain multiple sources. You cannot just provide a list of IP addresses.
`400`	`{"code":"010006", "name":"AddressPortAlreadyInUse", "message":"Address or port already in use", "httpStatusCode":400}`	The system is already streaming to the specified IP address or port.
`400`	`{"code":"010006", "name":"AddressPortAlreadyInUse", "message":"Address or port already in use by a source", "httpStatusCode":400}`	The system is already configured to use a Source with the specified IP address or port.
`400`	`{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}`	Supplied data was insufficient or incorrect.
`404`	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.

Get Stream Information for a Video

Gets information about the specified stream for a video

Authorizations: Recordings - STREAM

Requests

```
GET /apis/assets/:id/streams/:streamId
```

Response

Success

```
{
  "data": {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "5e33bc64-4ff7-47aa-a7d6-562b35a9c4aa"
  }
}
```

Error

Code	Payload	Description
404	`{"code":"040003", "name":"StreamNotFound", "message":"Stream not found", "httpStatusCode":404}`	Stream with UUID `streamId` was not found.

Delete a Stream for a Video

Deletes the specified stream.

Authorizations: Recordings - STREAM

Requests

```
DELETE /apis/assets/:id/streams/:streamId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040003", "name":"StreamNotFound", "message":"Stream not found", "httpStatusCode":404}`	Stream with UUID `streamId` was not found.

Thumbnail Assets

Topics include:

- Get a Thumbnail Image for a Video
- Get a Thumbnail Image for a Video Track
- Update the Thumbnail for a Video

Get a Thumbnail Image for a Video

Retrieves the poster (thumbnail) image for the specified video.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/thumbnail
```

Response

Success

```
Thumbnail image in JPEG format.
```

Error

Code	Payload	Description
404	`{"code":"040011", "name":"ThumbnailNotFound", "message":"Thumbnail not found", "httpStatusCode":404}`	The thumbnail is missing or is still being generated.
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.

Related Topics

- Get a Thumbnail Image for a Video Track

Get a Thumbnail Image for a Video Track

This API endpoint allows integrators to request a specific thumbnail. HMP uses a separate poster image and thumbnail for each track/source of a multi-source video.

Authorizations: Recordings - VIEW

Requests

```
GET /apis/assets/:id/tracks/:trackId/thumbnail
```

Response

Success

```
Thumbnail image in JPEG format.
```

Error

Code	Payload	Description
404	`{"code":"040011", "name":"ThumbnailNotFound", "message":"Thumbnail not found", "httpStatusCode":404}`	The thumbnail is missing or is still being generated.
404	`{"code":"040002", "name":"AssetNotFound", "message":"Video not found", "httpStatusCode":404}`	Video with UUID `Id` not found.
404	`{"code":"040015","name":"TrackNotFound" , "message":"Track not found", "httpStatusCode":404}`	The track could not be found.

Update the Thumbnail for a Video

Updates the poster image for the video. **Authorizations:** Recordings - EDIT

Requests

```
PUT /apis/assets/:id/thumbnail

{
  "data": {
    "timestamp": 1000,
    "trackId": 3
  }
}
```

Parameters

Property	JSON Type	Property Type	Description
timestamp	Number	Required	Time in milliseconds to update the thumbnail.
trackId	Number	Required	Track to use for the poster image.

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
400	`{code:"040002", "name":"AssetNotFound", "message": "Video not found": 400}`	Video with UUID `Id` not found.

Code	Payload	Description

404	`{"code":"040015","name":"TrackNotFound",`
	`"message":"Track not found",`
	`"httpStatusCode":404}`
		The track could not be found.

Public Share Assets

Topics include:

- Get a Public Share List for a Video
- Create a Public Share for a Video
- Delete a Public Share for a Video

Get a Public Share List for a Video

Gets a list of public shares created for the specified video (asset).

Authorizations: Recordings - SHARE

Requests

```
GET /apis/assets/:id/publicshare
```

Response

Success

```
{
  "data":{"permission":"WATCH"}
}
```

Error

Code	Payload	Description

404	`{"code":"040025",`
	`"name":"SharePermissionNotFound",`
	`"message":"Share permission not found",`
	`"httpStatusCode":404}`
		The specified share permission was not found.

Create a Public Share for a Video

Creates a public share for the specified video (asset).

Authorizations: Recordings - WATCH

Requests

```
POST /apis/assets/:id/publicshare
{
  id: "a5e21b0b-516b-42c9-90be-20a3eb5b2c21",
  resource: "assets",
  permission: "WATCH"
}
```

Response

Success

```
{
 "data":{"permission":"WATCH"}
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
401	{"code":"020001","name":"UserNotAuthorized", "message":"User is not authorized", "httpStatusCode":401}	User is not authenticated.
403	{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}	User is not allowed to share to this permission ID.
404	{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The specified share permission was not found.

Delete a Public Share for a Video

Deletes the specified public share for a video (asset).

Authorizations: Recordings - SHARE

Requests

```
DELETE /apis/assets/:id/publicshare
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidation", "message":<Error message>, "httpStatusCode":400}	Supplied data was insufficient or incorrect.
403	{"code":"030003", "name":"CannotShare", "message":"Cannot share", "httpStatusCode":403}	User is not allowed to share to this permission ID.
404	{"code":"040025", "name":"SharePermissionNotFound", "message":"Share permission not found", "httpStatusCode":404}	The specified share permission was not found.
404	{"code": "040021", "name":"UserGroupNotFound", "message":"User or Group not found", "httpStatusCode":404}	The specified user or group was not found.
404	{"code": "040004", "name":"UserNotFound", "message":"User not found", "httpStatusCode":404}	The user with provided `id` or `username` was not found.
404	{"code":"040006", "name":"AssetNotFound", "message":"Asset not found", "httpStatusCode":404}	Unable to locate asset.

Stream Resources

Synopsis

HMP videos may be re-streamed to a multicast or unicast IPv4 address in order to play streams back using hardware decoders or software tools, e.g., for large scale viewing.

The streams API is used to get information about streams as well as control (i.e., play, pause, stop, and seek) the re-streamed video.

Stream Parameters

Property	JSON Type	Property Type	Description
id	String	Immutable	Identifies the stream.
address	String	Required	The network address (IPv4 or domain) to stream to.

Property	JSON Type	Property Type	Description
port	Integer	Required	The network port (0–65535) to stream to.
username	String	Optional	Name of the user who started the stream.
asset	String	Optional	Video with which the stream is associated.
state	String	Optional	Optional state (“playing” or “paused”), defaults to “playing”.
destinations	Array of strings	Optional	Optional list of objects for multiple destinations. List contains “trackId”, “address”, and “port”.

Get the Stream List

This request gets a list of streams/active re-streams from HMP.

Authorizations: System - VIEW

Requests

```
GET /apis/streams
```

Response

Success

```
{
  "data": [
    {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "2ecc5663-06ee-44be-863a-2b32ef401c4d",
    "destinations": [{
      "trackId": 3,
      "address": "239.134.32.100",
      "port": 4900
    }]
    }
  ]
}
```

Error

Code	Payload	Description
404	{"code":"040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	No results returned from the server.

Get Stream Information

Gets information about the specified stream.

Authorizations: Recordings - STREAM

Requests

```
GET /apis/streams/:streamId
```

Response

Success

For a single destination, the “address” and “port” are preserved (for backwards compatibility):

```
{
  "data": {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "2ecc5663-06ee-44be-863a-2b32ef401c4d",
    "destinations": [{
      "trackId": 3,
      "address": "239.134.32.100",
      "port": 4900
    }]
  }
}
```

For a stream with multiple destinations, address will be set to “multiple” and port to 0.

```
{
  "data": {
    "address": "multiple",
    "port": 0,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "2ecc5663-06ee-44be-863a-2b32ef401c4d",
    "destinations": [{
      "trackId": 3,
      "address": "239.134.32.100",
      "port": 4900
    }, {
      "trackId": 4,
      "address": "239.134.32.101",
      "port": 4900
    }]
  }
}
```

Error

Code	Payload	Description
404	{"code":"040003", "name":"StreamNotFound", "message":"Stream not found", "httpStatusCode":404}	Stream with UUID `streamId` was not found.

Pause or Resume a Stream

Pauses or resumes a stream

Authorizations: Recordings - STREAM

Requests

State can be “playing” or “paused”. If “state” is omitted, then it is toggled.

```
PUT /apis/streams/:streamId

{
  "state": "playing"
}
```

Response

Success

```
{
  "data": {
    "address": "239.134.32.100",
    "port": 4900,
    "asset": "a907804b-d6d5-4328-9bd6-5aca9abbac8c",
    "username": "haiadmin",
    "id": "2ecc5663-06ee-44be-863a-2b32ef401c4d",
    "state": "playing",
    "destinations": [{
      "trackId": 3,
      "address": "239.134.32.100",
      "port": 4900
    }
  }
}
```

Error

Code	Payload	Description
400	{"code":"010001", "name":"InputValidationFailed", "message":"Invalid state <invalid_state>", "httpStatusCode":400}	Invalid state.
404	{"code":"040003", "name":"StreamNotFound", "message":"Stream not found", "httpStatusCode":404}	A stream with UUID `streamId` was not found.

Stop a Stream

Stops a stream.

Authorizations: Recordings - STREAM

Requests

```
DELETE /apis/streams/:streamId
```

Response

Success

```
(Empty payload)
```

Error

Code	Payload	Description
404	`{"code":"040003", "name":"StreamNotFound", "message":"Stream not found", "httpStatusCode":404}`	Stream with UUID `streamId` was not found.

Jump to a Point in a Stream

This request jumps to a particular point in the stream specified by a time in seconds.

Authorizations: Recordings - STREAM

Requests

```
POST /apis/streams/:streamId/seek

{
  "time": 60.5
}
```

Response

Success

```
{
  "data": {
    "time": 60.5
  }
}
```

Error

Code	Payload	Description

`400`	`{"code":"010001",`
	`"name":"InputValidationFailed",`
	` "message":"<message>",`
	`"httpStatusCode":400}`
		Invalid input.

`404`	`{"code":"040003","name":"StreamNotFound",`
	` "message":"Stream not found",`
	`"httpStatusCode":404}`
		A stream with UUID `streamId` was not found.

System Resources

In Version 2.5, the `system` API is used to get information about the HMP version.

Get the HMP Software/Build Version

Gets the software version and build number for HMP.

Authorizations: All

Requests

```
GET /apis/system/version
```

Response

Success

```
{
  "data": {
    "version": "2.0.0",
    "build": "30242"
  }
}
```

Error

None.

Event Resources

Synopsis

An *event* is a single or recurring occurrence scheduled to be streamed (i.e., live), recorded or both. As of Version 2.5, an HMP session is associated with an event. Live events can be simple single source streams or the broadcast of multiple sources simultaneously (sessions). Scheduled events can be shared with authenticated and/or non-authenticated users throughout an enterprise network.

The `events` API is used to generate a list of all the events for the system.

Tip

To generate a list of events for a specific session, see **Get an Event List**.

Event Parameters

Property	JSON Type	Property Type	Description
id	UUID	Required	Event ID
title	String	Optional	Session title
sessionId	UUID	Required	Session ID
startTime	Number	Required	Timestamp in seconds when event triggers (or first occurrence if type is weekly-recurring)
duration	Number	Optional	Duration in seconds
type	String	Required	“one-time” or “weekly-recurring”
command	String	Required	“set-live” or “record”
parameters	Object	Required when command=”set-live”	Parameters used with the command. For example, for command="set-live", a record parameter can be passed to indicate whether to record the session as well as set it live.

Get List of Events

Gets a list of events created for HMP.

Authorizations: Sessions - VIEW

Requests

```
GET /apis/events
```

Tip

You can query events based on a title and description (or HotMark title and description) by adding the “ `q` ” query parameter to the collection URL. For more information, see **Queries**.

Response

Success

```
{
  "data": [
    {
      "id": "96097207-ca7e-4764-93d7-41a1eedc5b84",
      "title": "Session - 10/21/2015, 2:04 PM",
      "sessionId": "99bdd65c-5493-4d5c-b583-b19336f18c8e",
      "startTime": 1445452200,
      "duration": 1800,
      "type": "weekly-recurring",
      "typeData": {
        "days": ["MON", "WED", "SAT"]
      },
      "command": "set-live",
      "parameters": {
        "record": true
      }
    },
    {
      "id": "151230d6-6e7e-4bdb-b3ed-824900507cb1",
      "title": "Session - 10/21/2015, 2:17 PM",
      "sessionId": "6fe27366-767b-4fd2-a89c-b30e1c7d7e59",
      "startTime": 1445456820,
      "duration": 89280,
      "type": "one-time",
      "command": "set-live",
      "parameters": {
        "record": false
      }
    }
  ],
  "paging": {
    "results": 2,
    "pageSize": 15
  }
}
```

Error

Code	Payload	Description
404	{"code": "040012", "name":"NoResults", "message":"No results", "httpStatusCode":404}	No results returned from the server.

Warranties

1-Year Limited Hardware Warranty

Haivision warrants its hardware products against defects in materials and workmanship under normal use for a period of ONE (1) YEAR from the date of equipment shipment ("Warranty Period"). If a hardware defect arises and a valid claim is received within the Warranty Period, at its option and to the extent permitted by law, Havision will either (1) repair the hardware defect at no charge, or (2) exchange the product with a product that is new or equivalent to new in performance and reliability and is at least functionally equivalent to the original product. A replacement product or part assumes the remaining warranty of the original product or ninety (90) days from the date of replacement or repair, whichever is longer. When a product or part is exchanged, any replacement item becomes your property and the replaced item becomes Haivision's property.

EXCLUSIONS AND LIMITATIONS

This Limited Warranty applies only to hardware products manufactured by or for Haivision that can be identified by the "Haivision" trademark, trade name, or logo affixed to them. The Limited Warranty does not apply to any non-Haivision hardware products or any software, even if packaged or sold with Haivision hardware. Manufacturers, suppliers, or publishers, other than Haivision, may provide their own warranties to the end user purchaser, but Haivision, in so far as permitted by law, provides their products "as is".

Haivision does not warrant that the operation of the product will be uninterrupted or error-free. Haivision does not guarantee that any error or other non-conformance can or will be corrected or that the product will operate in all environments and with all systems and equipment. Haivision is not responsible for damage arising from failure to follow instructions relating to the product's use.

This warranty does not apply:

(a) to cosmetic damage, including but not limited to scratches, dents and broken plastic on ports;

(b) to damage caused by accident, abuse, misuse, flood, fire, earthquake or other external causes;

(c) to damage caused by operating the product outside the permitted or intended uses described by Haivision;

(d) to a product or part that has been modified to alter functionality or capability without the written permission of Haivision; or

(e) if any Haivision serial number has been removed or defaced.

TO THE EXTENT PERMITTED BY LAW, THIS WARRANTY AND REMEDIES PROVIDED ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, REMEDIES AND CONDITIONS, WHETHER ORAL OR WRITTEN, STATUTORY, EXPRESS OR IMPLIED. AS PERMITTED BY APPLICABLE LAW, HAIVISION SPECIFICALLY DISCLAIMS ANY AND ALL STATUTORY OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND WARRANTIES AGAINST HIDDEN OR LATENT DEFECTS. IF HAIVISION CANNOT LAWFULLY DISCLAIM STATUTORY OR IMPLIED WARRANTIES THEN TO THE EXTENT PERMITTED BY LAW, ALL SUCH WARRANTIES SHALL BE LIMITED IN DURATION TO THE DURATION OF THIS EXPRESS WARRANTY AND TO REPAIR OR REPLACEMENT SERVICE AS DETERMINED BY HAIVISION IN ITS SOLE DISCRETION. No Haivision reseller, agent, or employee is authorized to make any modification, extension, or addition to this warranty. If any term is held to be illegal or unenforceable, the legality or enforceability of the remaining terms shall not be affected or impaired.

EXCEPT AS PROVIDED IN THIS WARRANTY AND TO THE EXTENT PERMITTED BY LAW, HAIVISION IS NOT RESPONSIBLE FOR DIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY BREACH OF WARRANTY OR CONDITION, OR UNDER ANY OTHER LEGAL THEORY, INCLUDING BUT NOT LIMITED TO LOSS OF USE; LOSS OF REVENUE; LOSS OF ACTUAL OR ANTICIPATED PROFITS (INCLUDING LOSS OF PROFITS ON CONTRACTS); LOSS OF THE USE OF MONEY; LOSS OF ANTICIPATED SAVINGS; LOSS OF BUSINESS; LOSS OF OPPORTUNITY; LOSS OF GOODWILL; LOSS OF REPUTATION; LOSS OF, DAMAGE TO OR CORRUPTION OF DATA; OR ANY INDIRECT OR CONSEQUENTIAL LOSS OR DAMAGE HOWSOEVER CAUSED INCLUDING THE REPLACEMENT OF EQUIPMENT AND PROPERTY, ANY COSTS OF RECOVERING, PROGRAMMING, OR REPRODUCING ANY PROGRAM OR DATA STORED OR USED WITH HAIVISION PRODUCTS AND ANY FAILURE TO MAINTAIN THE CONFIDENTIALITY OF DATA STORED ON THE PRODUCT. THE FOREGOING LIMITATION SHALL NOT APPLY TO DEATH OR PERSONAL INJURY CLAIMS, OR ANY STATUTORY LIABILITY FOR INTENTIONAL AND GROSS NEGLIGENT ACTS AND/OR OMISSIONS.

OBTAINING WARRANTY SERVICE

Before requesting warranty service, please refer to the documentation accompanying this hardware product and the Haivision Support Portal https://support.haivision.com. If the product is still not functioning properly after making use of these resources, please contact Haivision or Authorized Reseller using the information provided in the documentation. When calling, Haivision or Authorized Reseller will help determine whether your product requires service and, if it does, will inform you how Haivision will provide it. You must assist in diagnosing issues with your product and follow Haivision's warranty processes.

Haivision may provide warranty service by providing a return material authorization ("RMA") to allow you to return the product in accordance with instructions provided by Haivision or Authorized Reseller. You are fully responsible for delivering the product to Haivision as instructed, and Haivision is responsible for returning the product if it is found to be defective. Your product or a replacement product will be returned to you configured as your product was when originally purchased, subject to applicable updates. Returned products which are found by Haivision to be not defective, out-of-warranty or otherwise ineligible for warranty service will be shipped back to you at your expense. All replaced products and parts, whether under warranty or not, become the property of Haivision. Haivision may require a completed pre-authorized form as security for the retail price of the replacement product. If you fail to return the replaced product as instructed, Haivision will invoice for the pre-authorized amount.

APPLICABLE LAW

This Limited Warranty is governed by and construed under the laws of the Province of Quebec, Canada.

This Limited Hardware Warranty may be subject to Haivision's change at any time without prior notice.

EULA - End User License Agreement

READ BEFORE USING

THE LICENSED SOFTWARE IS PROTECTED BY COPYRIGHT LAWS AND TREATIES. READ THE TERMS OF THE FOLLOWING END USER (SOFTWARE) LICENSE AGREEMENT ("AGREEMENT") CAREFULLY BEFORE ACCESSING THE LICENSED SOFTWARE. BY SCANNING THE QR CODE TO REVIEW THIS AGREEMENT AND/OR ACCESSING THE LICENSED SOFTWARE, YOU CONFIRM YOUR ACCEPTANCE OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THESE TERMS, HAIVISION IS UNWILLING TO LICENSE THE LICENSED SOFTWARE TO YOU AND YOU ARE NOT AUTHORIZED TO ACCESS THE LICENSED SOFTWARE.

Click the following link to view the Software End-User License Agreement: **Haivision EULA.pdf**

If you have questions, please contact **legal@haivision.com**

SLA - Service Level Agreement

1. Introduction

This Service Level and Support supplement forms a part of and is incorporated into the Service Agreement (the “Agreement”) between You and Haivision Network Video Inc. (“Haivision”). Capitalized terms used but not otherwise defined in this supplement shall have the meaning ascribed to them in the Agreement. Haivision may, upon prior written notice to You, amend this supplement to incorporate improvements to the service levels and support commitments at no additional cost to You. This supplement applies only to those products and services set forth below.

2. Definitions

- "Audience Member" means an individual or entity that accesses Your Published Media Objects through a public URL.
- "Access Service" means the service provided by Haivision VCMS that verifies an Audience Member’s credentials.
- "Digital Media File" means a computer file containing text, audio, video, or other content.
- "Outage" is a 12-minute period of consecutive failed attempts by all six agents to PING the domain on the Haivision Streaming Media network.
- "Published Media Object" means a Digital Media File with a public URL.
- "Transaction" means the creation of a right for an Audience Member to access a Media Object and the completion of an order logged in the order history service.

3. Service Levels for the Video Content Management System

The service levels in this **Section 3** apply only to the hosted version of Haivision VCMS and the Haivision VCMS development kit (collectively, the “Standard Hosted Components” of Haivision Video Cloud Services). Subject to the exceptions noted in **Section 4** below, the aforementioned components of Haivision Video Cloud Services will be available for use over the course of each calendar month as follows:

Type of Access	Definition	Availability Level
Write Functions	• Access to all functions through the administrative user interface. • Ability to add or modify objects and metadata through the application programming interface ("API") • Ability of ingest service to check for new or updated files or feeds	99.999%
Read-Only Functions	• Ability to retrieve data through the API • Ability for Audience Members to authenticate through the Access Service • Ability for Audience Members to play Published Media Objects • Ability for Audience Members to play Haivision VCMS-authenticated or entitled Published Media Objects • Ability to complete Transactions	99.999%

4. Exceptions to Availability for the VCMS

The Standard Hosted Components may not be available for use under the following circumstances, and in such case such periods of unavailability shall not be counted against Haivision Video Cloud for purposes of calculating availability:

a. Normal Maintenance, Urgent Maintenance and Upgrades as defined in the table below;
b. Breach of the Agreement by You as defined in the Agreement;
c. The failure, malfunction, or modification of equipment, applications, or systems not controlled by Haivision Video Cloud;
d. Any third party, public network, or systems unavailability;
e. Acts of Force Majeure as defined in the Agreement;
f. Modification of software made available to You as part of Haivision Video Cloud Services by You or a third party acting on Your behalf; and
g. Any third party product or service not incorporated into Haivision Video Cloud Services or any third party plug-in.

Haivision Video Cloud shall make commercially reasonable efforts to notify, or work with, applicable third parties to repair or restore Haivision VCMS functionality affected by such exceptions.

Type of Maintenance	Purpose	Write Functions Available	Read Functions Available	Maximum Time Per Month	Continuous Time in Mode (Max)	Window (Central Time)	Min Notice
Normal	• Preventive maintenance on the software/ hardware components of Haivision VCMS • Addition of new features/functions • Repair errors that are not immediately affecting Your use of Haivision VCMS	No	Yes	10 Hours	6 Hours	10:00pm - 5:00am	48 Hours
Urgent	• Repair errors that are immediately affecting Your use of Haivision VCMS	No	Yes	30 Minutes	15 Minutes	Any Time	3 Hours

Type of Maintenance	Purpose	Write Functions Available	Read Functions Available	Maximum Time Per Month	Continuous Time in Mode (Max)	Window (Central Time)	Min Notice
Upgrades	• Perform upgrades on software or hardware elements necessary to the long term health or performance of Haivision VCMS, but which, due to their nature, require that certain components of Haivision VCMS to be shut down such that no access is possible	No	No	1 Hour	1 Hour	12:00am - 4:00am M-F	5 Days

5. Credits for Downtime for the VCMS

Haivision Video Cloud will grant a credit allowance to You if You experience Downtime in any calendar month and you notify Haivision Video Cloud thereof within ten (10) business days after the end of such calendar month. In the case of any discrepancy between the Downtime as experienced by You and the Downtime as measured by Haivision Video Cloud, the Downtime as measured by Haivision Video Cloud shall be used to calculate any credit allowance set forth in this section. Such credit allowance shall be equal to the pro-rated charges of one-half day of Fees for each hour of Downtime or fraction thereof. The term “Downtime” shall mean the number of minutes that Standard Hosted Components are unavailable to You during a given calendar month below the availability levels thresholds in **Section 3**, but shall not include any unavailability resulting from any of the exceptions noted in **Section 4**. Within thirty (30) days after the end of any calendar month in which Downtime occurred below the availability levels thresholds in **Section 3**, Haivision Video Cloud shall provide You with a written report detailing all instances of Downtime during the previous month. Any credit allowances accrued by You may be offset against any and all Fees owed to Haivision Video Cloud pursuant to the Agreement, provided that a maximum of one month of credit may be accrued per month.

6. Support Services for the VCMS

Support for Haivision Video Cloud Services as well as the Application Software (defined as the VCMS application software components that Haivision licenses for use in conjunction with the Video Cloud Services) can be reached at **hvc-techsupport@haivision.com** and shall be available for all Your support requests. Haivision Video Cloud will provide 24x7 monitoring of the Standard Hosted Components.

Cases will be opened upon receipt of request or identification of issue, and incidents will be routed and addressed according to the following:

Severity Level	Error State Description	Status Response Within	Incident Resolution within
1 – Critical Priority	Renders Haivision VCMS inoperative or causes Haivision VCMS to fail catastrophically.	15 minutes	4 hours
2 – High Priority	Affects the operation of Haivision VCMS and materially degrades Your use of Haivision VCMS.	30 minutes	6 hours
3 – Medium Priority	Affects the operation of Haivision VCMS, but does not materially degrade Your use of Haivision VCMS.	2 hours	12 hours

Severity Level	Error State Description	Status Response Within	Incident Resolution within
4 – Low Priority	Causes only a minor impact on the operation of Haivision VCMS.	1 business day	3 business days

7. Service Levels for Haivision Streaming Media Service

Haivision agrees to provide a level of service demonstrating 99.9% Uptime. The Haivision Streaming Media Service will have no network Outages.

The following methodology will be employed to measure Streaming Media Service availability:

Agents and Polling Frequency

a. From six (6) geographically and network-diverse locations in major metropolitan areas, Haivision's Streaming Media will simultaneously poll the domain identified on the Haivision Streaming Media network.
b. The polling mechanism will perform a PING operation, sending a packet of data and waiting for a reply. Success of the PING operation is defined as a reply being received.
c. Polling will occur at approximately 6-minute intervals.
d. Based on the PING operation described in (b) above, the response will be assessed for the purpose of measuring Outages.

If an Outage is identified by this method, the customer will receive (as its sole remedy) a credit equivalent to the fees for the day in which the failure occurred.

Haivision reserves the right to limit Your use of the Haivision Streaming Media network in excess of Your committed usage in the event that Force Majeure events, defined in the Agreement, such as war, natural disaster or terrorist attack, result in extraordinary levels of traffic on the Haivision Streaming Media network.

8. Credits for Outages of Haivision Streaming Media Service

If the Haivision Streaming Media network fails to meet the above service level, You will receive (as your sole remedy) a credit equal to Your or such domain's committed monthly service fee for the day in which the failure occurs, not to exceed 30 days of fees.

9. No Secondary End User Support

UNDER NO CIRCUMSTANCES MAY YOU PROVIDE CONTACT INFORMATION FOR HAIVISION SERVICES TO CUSTOMERS OR AUDIENCE MEMBERS OR OTHER THIRD PARTIES WITHOUT HAIVISION'S EXPRESS PRIOR WRITTEN CONSENT.

Getting Help

General Support	North America (Toll-Free) **1 (877) 224-5445** International **1 (514) 334-5445** *and choose from the following:* Sales - 1, Cloud Services - 3, Support - 4
Managed Services	U.S. and International 1 (512) 220-3463
Fax	1 (514) 334-0088
Support Portal	https://support.haivision.com
Product Information	info@haivision.com

info@haivision.com
www.haivision.com

North America: 1.877.224.5445
International: +1.514.334.5445

HAIVISION