About the Wowza Video REST API

Updated on 08/10/2022 2:58 pm PDT

Learn about the REST API for the Wowza Video™ service, including setting up a request, versioning, authentication, and testing out the API.

About the REST API

The Wowza Video REST API (application programming interface) uses the Hypertext Transfer Protocol (HTTP) to request data from Wowza Video servers through requests to API endpoints. You can use the Wowza Video REST API to add live streaming and playback functionality to your applications. The REST API offers complete programmatic control over live streams, transcoders, stream sources, and stream targets. Anything you can do in the Wowza Video UI can also be achieved by making HTTP-based requests to cloud-based servers through the REST API.

API requests

Requests in the Wowza Video REST API use JSON syntax for the request body and response. The examples in our articles use a **curl** command to execute an HTTP method in a Command Prompt or Terminal window. To learn about cURL and other methods for testing out the REST API, see Tools for testing the API.

Curl commands use the following general format:

```
curl -[HTTP method] --[headers] -[parameters] "[resource]"
```

HTTP method is the action you're requesting of the resource. Wowza Video uses these HTTP methods:

HTTP method	Description
POST	POST creates records in the resource's database. The POST method returns a response that indicates that the request was successful and includes the values of any newly created records.
GET	GET retrieves records from the resource's database. The GET method returns a response including detailed information about the queried object or objects. You may need values from a GET response in order to use them in a POST or PATCH request. Some endpoints return a cached response if you make a duplicate GET request to the endpoint within a specified period of time. See Wowza Video REST API server response caching for more information.

HTTP method	**Description**
PATCH	PATCH updates records in the resource's database. Certain attributes can't be updated in a PATCH request and must be defined in an initial POST request. The PATCH method returns a response that indicates that the request was successful and includes the values of updated records.
PUT	PUT requests in the Wowza Video REST API start or stop a entity such as a live stream or transcoder. The PUT method response gives information about the entity that was affected.
DELETE	DELETE removes the record from the resource's database. The delete method returns a response with no content.

Headers are information that precede the actual HTTP request. Wowza Video REST API requests require headers in key:value pairs.

Key	**Description**	**Where used**
Authorization	Bearer [your authentication token] Authenticates your API requests. See Authentication.	All requests.
Content-Type	application/json Specifies that the content sent to the server is in JSON format.	For POST and PATCH HTTP methods only.

Parameters refine the request and may correspond to options in the Wowza Video user interface. They can also be known as attributes. In JSON syntax, you define a value for each parameter in key:value pairs.

Example of parameters and their assigned values in a JSON object:

```
{
  "transcoder": {
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_central_iowa",
    "delivery_method": "push",
    "name": " MyPassthruTranscoder",
    "protocol": "rtmp",
    "transcoder_type": "passthrough"
  }
}
```

Resource, or base URL, is the location of the server receiving the request. In the Wowza Video REST API, the base URL is

https://api.video.wowza.com/api/[*version*]/

Example of a full API request in curl command syntax:

```
curl -X POST \
-H "Content-Type: application/json" \
-H "Authorization: Bearer ${WV_JWT}" \
-d '{
  "transcoder": {
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_central_iowa",
    "delivery_method": "push",
    "name": " MyPassthruTranscoder",
    "protocol": "rtmp",
    "transcoder_type": "passthrough"
  }
}' "https://api.video.wowza.com/api/v1.8/transcoders"
```

> **Note:** The Wowza Video REST API publishing and playback endpoints use dynamic IP addresses. As a result, we can't provide IP addresses for allowlisting from behind a firewall.

API versions

The Wowza Video REST API is periodically versioned. Minor updates are iterated using sequential dot numbers; major versions are iterated using sequential whole numbers. To learn more about the API lifecycle and what you can expect in each stage (beta, current, etc), see Wowza Video REST API lifecycle management.

More resources

- See the Wowza Video REST API deprecation policy for more information about deprecation and the deprecation schedule.
- See Wowza Video REST API release notes for a detailed changelog of updates to each API version.

API limits

Reasonable resource limits are in place to prevent excessive usage of streams and stream targets through the Wowza Video REST API. See Wowza Video REST API limits for details.

For information on limitations of the Wowza Video free trial, see Trial limitations.

Authentication

For security, all requests to the Wowza Video REST API must include authentication information in the header. Wowza Video uses JSON Web Token-based authentication, an open standard (RFC 7519) used for quickly and securely transmitting information in JSON format. You'll create an access token, or a JWT, in our Token Management portal and use it in your API requests.

> **Note:** Wowza Video is transitioning to the JSON Web Token (JWT)-based authentication scheme. The JWT-based authentication is available starting in version 1.8 of the API alongside the older API

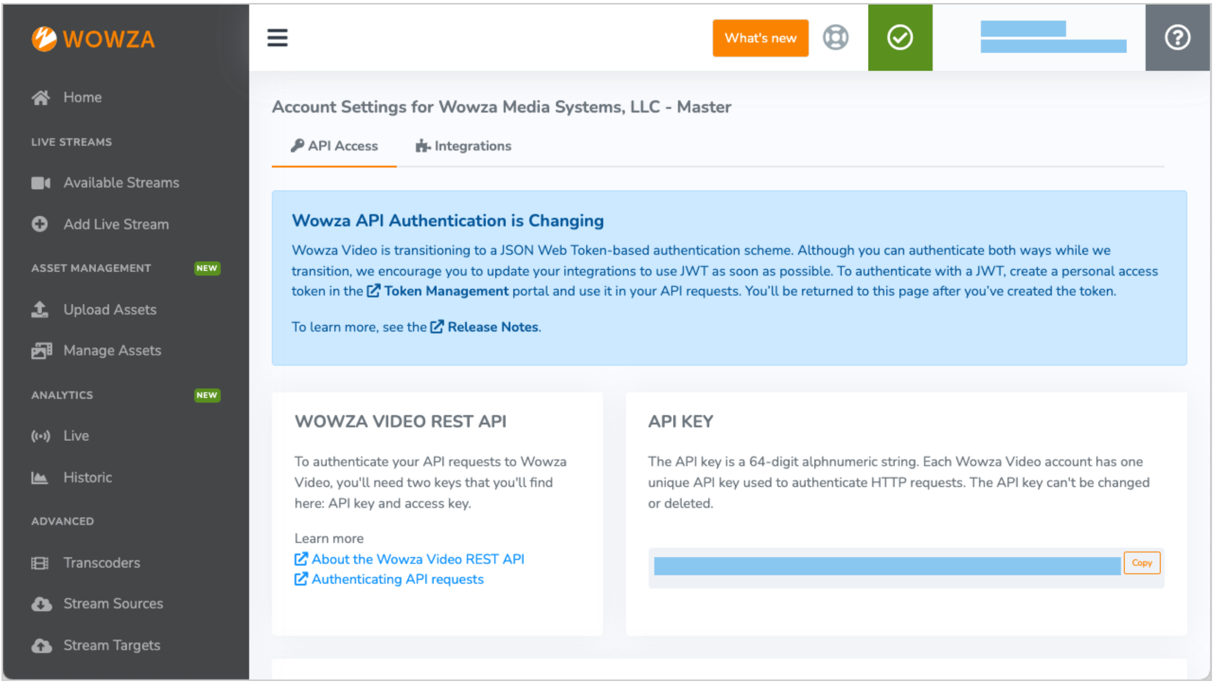
> key/access key and HMAC authentication schemes. However, version 1.9 and future versions of the API will only support JWT, and we encourage you to update your integrations to use JWT as soon as possible.
>
> All help content shows code samples with JWT-based authentication.

1. Generate an access token (JWT)

You can generate access tokens, also called JWTs in API authentication, in the Wowza Video user interface. There are two types of access tokens:

- **Personal access tokens** — Available to all users with a Wowza Video license. These tokens are specific to a user and can be used in API calls to do quick tests or manual actions that don't need to persist if the user leaves the organization.
- **System access tokens** — Available to organization owners. These tokens are specific to the organization and transcend individual users who come into and leave an organization. For this reason, system access tokens are best used for integrations that go into production environments and need to persist after specific users leave an organization.

1. In the menu bar, click your user name and choose **Account Settings**.
2. Make sure you're on the **API Access** tab, and then click **Token Management** in the **Wowza API Authentication is Changing** banner.

   [redacted]

   Account Settings for Wowza Media Systems, LLC - Master

   Wowza API Authentication is Changing

   Wowza Video is transitioning to a JSON Web Token-based authentication scheme. Although you can authenticate both ways while we transition, we encourage you to update your integrations to use JWT as soon as possible. To authenticate with a JWT, create a personal access token in the **Token Management** portal and use it in your API requests. You'll be returned to this page after you've created the token.

   To learn more, see the **Release Notes**.

   WOWZA VIDEO REST API

   To authenticate your API requests to Wowza Video, you'll need two keys that you'll find here: API key and access key.

   Learn more
   About the Wowza Video REST API
   Authenticating API requests

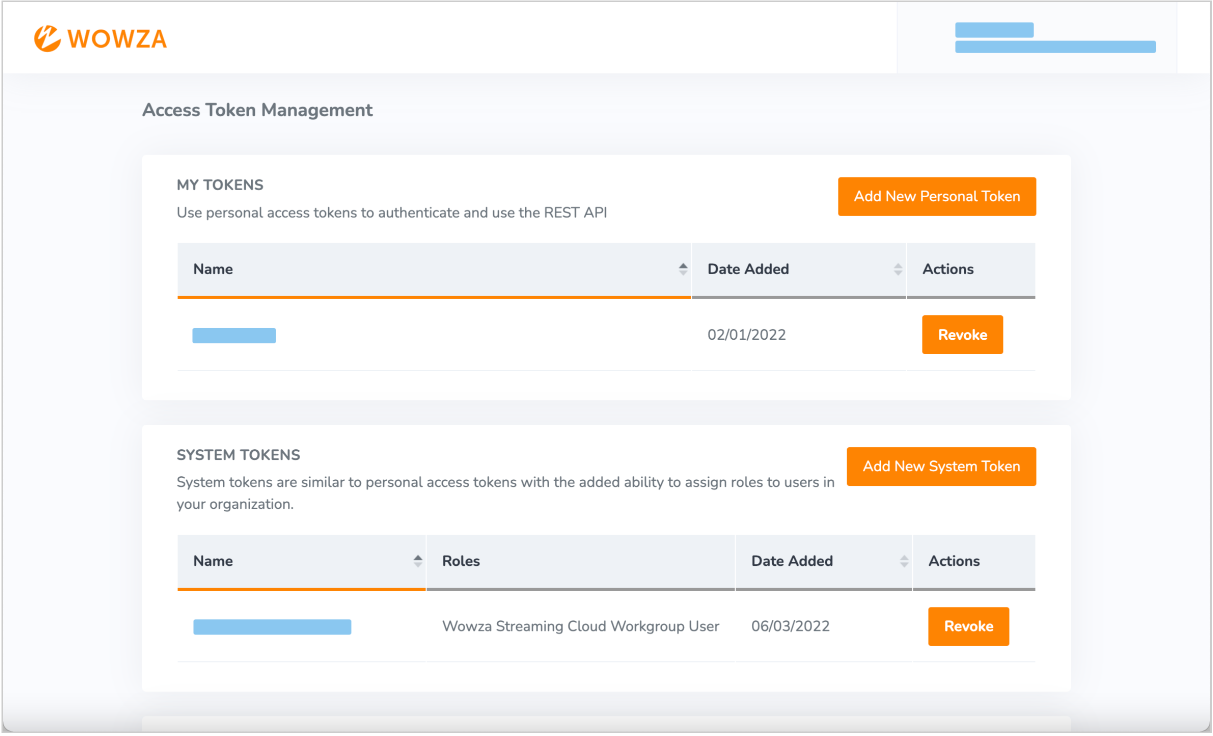
   API KEY

   The API key is a 64-digit alphnumeric string. Each Wowza Video account has one unique API key used to authenticate HTTP requests. The API key can't be changed or deleted.

   [redacted]

3. In the Token Management portal, add a new token. Choose a personal access token or, if you're an organization owner, a system token. If you're an organization member who needs a system-level token that needs to persists, reach out to your organization owner.

   Make sure you copy the token and store it in a safe location; you won't see the token again later.



Access Token Management

MY TOKENS

Use personal access tokens to authenticate and use the REST API

Name	Date Added	Actions
[redacted]	02/01/2022	Revoke

SYSTEM TOKENS

System tokens are similar to personal access tokens with the added ability to assign roles to users in your organization.

Name	Roles	Date Added	Actions
[redacted]	Wowza Streaming Cloud Workgroup User	06/03/2022	Revoke

If you revoke a token, you'll need to update any integrations that use it to access the Wowza Video REST API.

2. Use JWT authentication

To use the JWT you generated in the Token Management portal, send it as a bearer token in an Authentication header of your API requests.

A request using the JWT authentication method looks like this:

```
curl -X GET \
-H "Authorization: Bearer ${WV_JWT}" \
"${WV_HOST}/api/${WV_VERSION}/live_streams"
```

> **Important:** In example requests shown in our example articles, we use environment variables for the JWT, host, and version number. See Using cURL to learn more.

Tools for testing the API

Using cURL

The Wowza Video REST API examples in this documentation site use curl commands. cURL is a command-line tool that allows you to execute HTTP requests. It is native to the Terminal application on macOS and Linux, but it requires some installation for use in the Command Prompt on Windows operating systems. To find a download for Windows, see the curl Download Wizard.

To review curl command syntax used in our API examples, see API requests.

We use environment variables in the curl API request examples for Wowza Video to make it easier for you to copy, paste, and run commands in your Terminal or Command Prompt window. The variable syntax differs according to your operating system.

Description	**macOS/Linux Variable (used in examples)**	**Windows Variable**
JSON Web Token	${WV_JWT}	%WV_JWT%
Host	${WV_HOST}	%WV_HOST%
API version	${WV_VERSION}	%WV_VERSION%

You can set environment variables like this, depending on your operating system:

Linux or macOS:

```
export WV_JWT="your token here"
export WV_HOST="https://api.video.wowza.com"
export WV_VERSION="v1.8"
```

Windows:

```
set WV_JWT=your token here
set WV_HOST=https://api.video.wowza.com
set WV_VERSION=v1.8
```

After setting values for the environment variables, you can copy and paste API request examples into your Terminal or Command Prompt. You'll still need to substitute resource IDs and update parameter values, where appropriate.

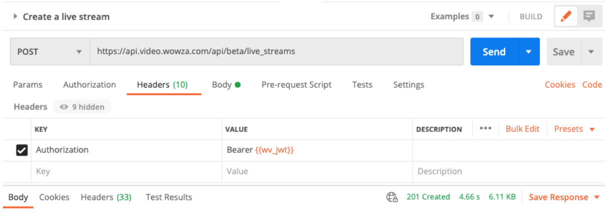
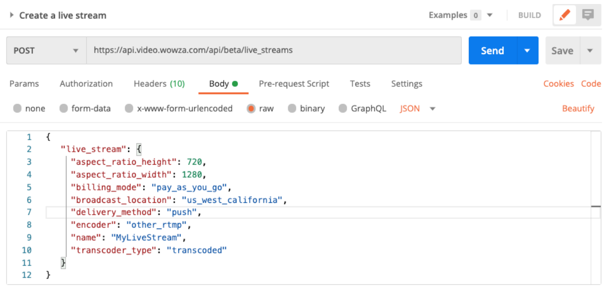
Using a GUI-based REST API client

Other options for testing the Wowza Video REST API are applications with a graphical user interface, such as Postman, Paw, or Insomnia, that provide a user interface and responses in formatted JSON. See documentation for your chosen API testing application for detailed usage information.

Note these configuration details:

Authorization	No Auth
Headers	Authorization: Bearer [your JWT] Content-Type: application/json
Body	JSON-formatted request body that includes parameters and their assigned values

Example POST request in a GUI-based API test application:

Headers	Body
	

POST https://api.video.wowza.com/api/beta/live_streams

KEY	VALUE	DESCRIPTION
☑ Authorization	Bearer {{wv_jwt}}	
Key	Value	Description

201 Created 4.66 s 6.11 KB

POST https://api.video.wowza.com/api/beta/live_streams

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
  "live_stream": {
    "aspect_ratio_height": 720,
    "aspect_ratio_width": 1280,
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_west_california",
    "delivery_method": "push",
    "encoder": "other_rtmp",
    "name": "MyLiveStream",
    "transcoder_type": "transcoded"
  }
}
```

Accessing support

Sometimes you need a helping hand. You can view the Wowza Video REST API community forum to get help from other Wowza Video users and experts from around the world.

You can also contact Wowza Support for REST API issues with Wowza Video. When contacting support for an error returned via the API, be sure to include the following information:

- The API request you sent that caused an error
- The request ID returned in the error response
- The request timestamp returned in the error response

The request ID and request timestamp will look something like this:

```
{
   "meta": {
      "status": 401,
      "code": "ERR-401-InvalidSignature",
      "title": "Invalid Signature Error",
      "message": "Invalid signature.",
      "description": ""
   },
   "request_id": "12a6c19ad7fa49dc9126e6ce6c7ca9d7",
   "request_timestamp": "2019-11-04T15:38:47.198Z"
}
```

More resources

- Wowza Video REST API release notes
- Wowza Video REST API reference documentation (current version)
- Start live streaming using the Wowza Video REST API