Get filtered query results with the Wowza Streaming Cloud REST API

Originally Published on 02/26/2018 | Updated on 12/12/2019 9:09 am PST

Learn how to get filtered results from REST API queries made in the Wowza Streaming Cloud™ service.

Using the filter parameter

To execute a request that returns filtered results, use the filter parameter in the query. The filter parameter restricts the data that gets returned to one or more values associated with a field.

You can use the filter parameter with any GET call to the transcoders endpoint:

https://api.cloud.wowza.com/api/[version]/transcoders

To use the filter parameter, append the query URL with a two-part expression that specifies the field on which to filter and the logic (comparison operator) to use to filter. Use the syntax

?filter[n][field]=value&filter[n][comparison operator]=value

where

n is a zero-based index.

You can use as many filters as you want, combining them with an ampersand (&). Order doesn’t matter; multiple filters are additive.

Filter fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>created_at</td>
<td>The date and time the transcoder was created. Format as YYYY-MM-DDTHH:MM:SS.000Z where HH is a 24-hour clock in UTC.</td>
</tr>
</tbody>
</table>
Note: You must specify the exact creation date and time. Filtering on date ranges is currently not supported.

<table>
<thead>
<tr>
<th>id</th>
<th>The unique, eight-character alphanumeric string that identifies the transcoder.</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The descriptive name of the transcoder.</td>
</tr>
<tr>
<td>state</td>
<td>The state of the transcoder. Valid values are starting, stopping, started, stopped, and resetting.</td>
</tr>
</tbody>
</table>

Filter comparison operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>eq</td>
<td>Equals. Accepts one value. Returns only data whose field equals that value.</td>
</tr>
<tr>
<td>in</td>
<td>Equals any of. Accepts a comma-separated string of values. Returns any item whose field includes any of those values.</td>
</tr>
</tbody>
</table>

Example requests

Notes:
- To authenticate API requests, use HMAC authentication for production environments. For testing or proof of concept purposes only, use API key and access key authentication.
- The curl examples below use environment variables. See Using cURL for more information on how to set these up.
- If you’re using a request tool that has a globbing parser, turn off the parser so that the brackets don’t generate an error. In cURL, pass the `-g` flag with the request.
Return only transcoders whose state is **stopped**:

```
curl -X GET -g \
-H "wsc-api-key: ${WSC_API_KEY}" \
-H "wsc-access-key: ${WSC_ACCESS_KEY}" \
"${WSC_HOST}/api/${WSC_VERSION}/transcoders/?filter[0][field]=state&filter[0][eq]=stopped"
```

Return all transcoders that aren’t actively running (all transcoders whose state isn’t **started**):

```
curl -X GET -g \
-H "wsc-api-key: ${WSC_API_KEY}" \
-H "wsc-access-key: ${WSC_ACCESS_KEY}" \
"${WSC_HOST}/api/${WSC_VERSION}/transcoders/?filter[0][field]=state&filter[0][in]=starting,resetting,stopping,stopped"
```

Return only transcoders that are started and have the name **MyTranscoder** or **MyOtherTranscoder**:

```
curl -X GET -g \
-H "wsc-api-key: ${WSC_API_KEY}" \
-H "wsc-access-key: ${WSC_ACCESS_KEY}" \
"${WSC_HOST}/api/${WSC_VERSION}/transcoders/?filter[0][field]=state&filter[0][eq]=started&filter[1][field]=name&filter[1][in]=MyTranscoder,MyOtherTranscoder"
```