Making an API request

To generate a License Statistics report of your choosing, you can use any browser or an HTTP client (cURL). cURL is a library and command-line tool that can be used to access HTTP-based web services and websites.

Basics

Every License Statistics report defines a set of columns, each of which represents a possible field of a report row.

Columns that are invisible by default can be shown using the parameters described below.

Columns can be of various types, as described in Field types.

Request

Use the GET method to pass report parameters in a query string. Authenticate the request using a generated token.

Example request

```
curl -H "X-Auth-token: token" "http://yourdomain/url-to-api?parameter=value&parameter2=other_value"
```

where "yourdomain" is the domain where License Statistics is installed.

> ⓘ Note that if your installation uses either a port different than 80, or port 443 for SSL-secured HTTPS, you will need to provide the port number as well.

Common parameters

Format

The format type for reports is determined by the last segment of the URL, which can be json, csv, xslx or pdf. See the Response section below for details.

Report name

For PDF and XLSX formats, the name of the report can be set with parameter "rn".

Filtering

To filter out unwanted records, use the "filter" parameter. It takes JSON array of objects:

```
{
  "property": "${column_handle}",
  "operator": "${operator}",
  "value": "${operator_value}"


}
```

where:

- "property" specifies the column to filter by
- "operator" specifies the operator for filtering (see types of filters below)
- "value" specifies the value to use for the operator (see types of filters below)

String filter

Show only row where column ${column_handle} contains substring ${operator_value}

This filter is case-insensitive.

For example, if you specify a filter where the license server name (lsn) contains "asia", the matches would include "Asia server", "Asian server", "Caucasian server", etc.

```
filter=[{"property":"lsn","value":"asia"}]
```

Numeric, date, date and time filters

This filter lets you either specify equality (default) or a range operator, but not both.

Operator	Number	Date	Date and time
eq	equal to ${operator_value}		during day ${operator_value}
lt	less than ${operator_value}	before day ${operator_value}	before hour 00:00 of day ${operator_value}
gt	less than ${operator_value}	after day ${operator_value}	after hour 23:59 of day ${operator_value}

Example 1: Hour used (hu) between e and will match 2.72, 3 and 3.14, but will not match 2.7183 or 3.1415.

```
filter=[{"property":"hu","operator":"gt","value":"2.7183"},{"property":"hu","operator":"lt","value":"3.1415"}]
```

Example 2: Expiration date (fexp) after 1st of April 2019 (on 2nd of April or later).

```
filter=[{"property":"fexp","operator":"gt","value":"2019-01-04"}]
```

Example 3: Checkout time (lco) before 23rd of March 2019 (checked out on 22nd of March or earlier).

```
filter=[{"property":"lco","operator":"lt","value":"2019-03-23"}]
```

Boolean filter

Show only rows where flag (boolean column) ${column_handle} has value ${operator_value}.

Example: User is imported from LDAP (uil).

```
filter=[{"property":"uil","value":true}]
```

Enumeration filter

Show only rows where column ${column_handle} has one of the values from ${operator_value}.

Example: Features that are floating or token (all but node-locked).

```
filter=[{"property":"ftype","value":["Floating","Token"]}]
```

Column selection

By default, the report shows a subset of columns in left-to-right order. To show invisible columns, hide visible columns, or reorder columns, use the "fieldVisibility" parameter. This works for CSV, XLSX and PDF endpoints.

This takes a JSON object where keys are column, and value is string "1".

Example: to show columns lsid, lsn, fid, fns in order (from left to right):

```
fieldVisibility="{"lsid":"1","lsn":"1","fid":"1","fns":"1"}
```

Limiting and ordering results

You can order results using the "order" parameter and limit record count using the parameters "offset" and "limit." Order is applied first, then offset and limit.

- order - list of columns and directions to order by
- offset - rows to skip at the beginning of sorted result
- limit - maximum number of rows to return

```
order: [{"property":"vn","direction":"DESC"},{"property":"lsn","direction":"ASC"}]
offset: 9
limit: 5
```

Response

The API can respond with the following codes.

- 200 - everything OK
- 202 - data under preparation, used for historic reports to indicate that intermediate data (so called *aggregates*) is not ready yet
- 400 - bad parameter, refer to response body for details
- 401 - bad token or lack of, check if proper header was sent
- 402 - indicates problem with license
- 403 - such request is forbidden; check if you have permission to access report object (e.g., check that server or feature is not hidden)
- 500 - bug in server
- 503 - server is not yet ready

Body of response

JSON

If the URL ends with /json, the following object will be returned:

```
{"success": false, "msg": "${message}", "data": [], "totalCount": ${number} }
```

where:

- success - true if report was generated, otherwise false
- msg - short description of what went wrong
- data - array of objects, each representing report row (record)
- totalCount - number of rows that would be returned without applying "offset" and "limit", but still applying other parameters (including "filter")

Export to CSV, XLSX or PDF

If the URL ends with /csv, /xlsx or /pdf it will return a file in the corresponding format, as described below.

Type	MIME type	Description
csv	text/csv	Semicolon-separated values. Records separated by Windows line separator (CRLF)
xlsx	application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	Microsoft Office 2007 spreadsheet format
pdf	application/pdf	Adobe PDF format