Expiring Features API

This page refers to functionality that has been removed from License Statistics. This information no longer applies to License Statistics v6.3 and newer.

The Expiring Features endpoint delivers license usage metrics specific to the Expiring Features report for features that are nearing expiration.

Obtaining information about expiring features

You can easily list features that are nearing expiration by sending the following HTTP request.

```
GET /api/v1/report/expiry/server/${licenseServerId}/${returnType}?sd=${YYYY-MM-DD}&ed=${YYYY-MM-DD}$(additional parameters, as needed)
```

where $ indicates a variable value that you can replace with a value that best suits your needs. The possible parameters are described below.

Parameter	Required	Type	Description
${licenseServerId}	Yes	integer /string	Internal License Statistics identification of the license server for which you want to gather data about features nearing expiration. You can set this parameter to '*all*' to gather data from all license servers.
${returnType}	Yes	string	Standard format option. See Making an API request for details.
sd	No	date	Start date for which the report will be generated. If neither this field nor "ed" is provided, the date defaults to the current date.
ed	No	date	End date for which the report will be generated. If neither this field nor "sd" is provided, the date defaults to the current date + 12 months.
standard report options	No	various	See Making an API request for details.

Response

On success, this report will contain one row for each **feature** expiring in the specified time range. Each row consists of the following columns.

Column	Full name	Type	Description	Visible by default in export
fid	Feature ID	integer	Internal License Statistics identification of the feature.	✓ Yes
fns	Feature Name	string	Feature name.	✓ Yes
fv	Feature Version	enumeration	Feature version.	✓ Yes
ftype	Feature Type	string	See feature type descriptions for details.	✓ Yes
vn	Vendor	string	Name of license server vendor hosting the feature.	✓ Yes
lsid	License Server ID	integer	Internal License Statistics identification of license server hosting the feature.	✓ Yes
lsn	License Server	string	Name of license server hosting the feature.	✓ Yes
fexp	Feature Expires	date	Day when the feature expires.	✓ Yes

Note that the order in the table is the default order of the columns in the exported file.

Example 1

The following example shows a command that obtains statistics about all features expiring in the next 12 months on all license servers.

```
curl -H "X-Auth-token: token" "http://yourdomain/api/v1/report/expiry/server/all/json"
```

Example 2

The following example shows a command that obtains information about all features expiring in a specified period of time on a license server with an ID of "1."

```
curl --data "sd=2015-01-01" --data "ed=2015-03-01" -H "X-Auth-token: token" "http://yourdomain/api/v1/report/expiry/server/1/json"
```

Obtaining information about the number of features expiring in some period of time

You can easily list features that are going to expire in some period of time by sending the following HTTP request:

```
GET /api/v1/report/expiry/server/{licenseServerId}/count/grouped/json?sd=${YYYY-MM-DD}&ed=${YYYY-MM-DD}
&granularity=${granularity}&filter=${filter}
```

where the possible parameters can be explained as follows:

Parameter	Required	Type	Description
licenseServerId	Yes	integer/string	Internal License Statistics identification of license server for which you want to gather data about features nearing expiration. You can set this parameter to '*all*' to gather data from all license servers.
sd	No	date	Start date for which the report will be generated.If neither this field nor "ed" is provided, the date defaults to the current date.
ed	No	date	End date for which the report will be generated. If neither this field nor "sd" is provided, the date defaults to the current date + 12 months.
granularity	No	enumeration	Period type the result is aggregated for. HOUR granularity is not supported. See granularity descriptions for further details. By default, this parameter is set to MONTH
filter	No	JSON	See Making an API request for details.

This report supports only JSON format.

This report supports filtering, but does not support other standard options (limiting or ordering).

Response

On success, this report will contain one row for each **feature** expiring in the specified time range. Each row consists of the following columns.

Column	Full name	Type	Description
fexpy	Year	integer	Year for which the time period number has been specified.
fexpm	Period Number	integer	Number of time period such as day, week, month, etc.
fexpc	Count	integer	Count of features expiring in the given period of time.
granularity	Granularity	enumeration	Granularity given in parameter. Defaults to MONTH if unspecified.

Example 1

The following example shows a command that obtains a count of features that will expire for every month in the next 12 months on all license servers.

```
curl -H "X-Auth-token: token" "http://yourdomain/api/v1/report/expiry/server/all/count/grouped/json"
```

Example 2

The following example shows a command that obtains a count of features that will expire for every week in a specified period of time on a license server with an ID of "1."

```
curl --data "sd=2015-01-01" --data "ed=2015-03-01" --data "granularity=4" -H "X-Auth-token: token"
"http://yourdomain/api/v1/report/expiry/server/1/count/grouped/json"
```