

# Snyk Analytics Export API

Snyk Analytics Export API enables seamless data export by allowing you to initiate and manage export processes that generate CSV files and deliver them securely to a Snyk-managed storage. Designed for efficiency and security, the API supports exporting large datasets in an organized, scalable manner, making it ideal for reporting and analytics workflows.

This page presents the API specification and necessary parameters to configure the export process. You can use the API to export the Snyk issues dataset in the scope of Snyk Organization or Group. Navigate to the [available columns and filters](#) section to see the full lists.

## Consuming data process

The API includes three endpoints per scope, where the scope can be Snyk Organization or Snyk Group. Use the following workflow to successfully run an export using the API.

### 1. Initiate the export

Start by initiating an export process. The response to that request will return the `export_id`. Set the [filters and columns](#) according to your preferences.

 **POST** /groups/{group\_id}/export

### 2. Validate the export status

Proceed with validating the status using the export status endpoint and the `export_id` that was returned in the previous step.

There are three possible statuses:

- PENDING - the export process is running
- FINISHED - the export process has completed successfully. If the process finished successfully when this request arrives, the results data will be included in the response.
- ERROR - the export process has failed

 **GET** /groups/{group\_id}/jobs/export/{export\_id}

### 3. Fetch results

After the export process returns the finished status, fetch the exported files using the export result endpoint. Use the `export_id` that was returned in the first step.

# API Specification

API Server: <https://api.snyk.io/rest/>

- Group API endpoints specification
  - [POST Initiate an export process](#)
  - [GET Validate export status](#)
  - [GET Fetch export results](#)
- Org API endpoints specification
  - [POST Initiate an export process](#)
  - [GET Validate export status](#)
  - [GET Fetch export results](#)

## Rate Limits

The API rate limit is based on the actual data consumption versus the quantity of API calls and allows for the consumption of up to 5 million daily records.

## Data Retention

The exported CSV files will remain available in the designated S3 bucket for a period of three days.

## Available Columns and Filters


### Default filters

If the API call does not include a definition for the specific filters, the returned data is scoped by default to fetch issues that were introduced in the last 24 hours.

### Default Columns

If the API call does not define the specific columns, the returned data will include all the available columns by default.

# Available filters



Even though the requested filters themselves are not case sensitive, the filters' value is case sensitive. Use the exact filter value as it appears in the Snyk Web UI.

To clarify this requirement, the relevant filters in the available filters table are indicated with a case-sensitive indication next to them.

Filter	Description
introduced_date	The issue's introduction date. Acceptable format: YYYY-MM-DDTHH:MM:SSZ (example: 2024-11-28T09:10:00Z)
orgs	Snyk Org ID (available only for the group endpoints)
environment	The project's environment (case sensitive)
lifecycle	The project's lifecycle (case sensitive)

# Available columns

Column name	Description
group_public_id	A universally unique identifier for a group, assigned in the record's source database.
org_public_id	A universally unique identifier for an organization, assigned in the record's source database.
project_public_id	A universally unique identifier for a project, assigned in the record's source database.
problem_id	Snyk Vulnerability database ID that uniquely identifies the vulnerability.
product_name	The Snyk Product which initially identified this issue.
problem_title	Name of the Snyk discovered vulnerability.
vuln_db_url	URL which directs to the Snyk's Public Vulnerability Database website.
issue_type	Indicates whether the issue is related to a vulnerability, license, or configuration.
issue_sub_type	A more granular variation of issue type.
issue_url	URL which directs to the given project's instance of this vulnerability on the Snyk Website.

issue_status	Indicates whether the issue is open, resolved, or ignored.
issue_severity	Indicates the assessed level of risk, as Critical, High, Medium, or Low.
score	A score based on an analysis model. Priority score is released in General Availability, while Risk Score is in Early Access.
cve	Mitre CVE ID(s)
cwe	Mitre CWE ID(s)
exploit_maturity	Represents the existence and maturity of public exploits validated by Snyk, for example, Mature, Proof of concept.
introduction_category	A Snyk generated classification describing the nature of an issue's introduction in the context of Snyk product usage, for example, Baseline Issue, Non Preventable Issue, Preventable Issue.
snyk_cvss_score	Snyk's recommended Common Vulnerability Scoring System (CVSS) score.
snyk_cvss_vector	The vector string of the metric values used to determine the CVSS score.
nvd_severity	The vulnerability's severity as rated by NVD.
nvd_score	The vulnerability's score as calculated by NVD.
epss_score	The probability of exploitation in the wild in the next 30 days.
epss_percentile	The proportion of all vulnerabilities with the same or lower EPSS score.
computed_fixability	Indicates whether the issue can be fixed based on the vulnerability remediation paths.
fixed_in_available	Is the given vulnerability fixed in a different version of responsible source.
fixed_in_version	The first version in which a given vulnerability was fixed.
package_name_and_version	The vulnerability's associated package name and version.
semver_vulnerable_range	The vulnerable range of package versions (based on semantic versioning).
vulnerability_publication_date	The date a given vulnerability was first published by Snyk.
first_introduced	The timestamp of the first scan that identified the issue.
last_introduced	The most recent instance of an issue having been introduced (or reintroduced).

<code>last_ignored</code>	The most recent instance of an issue having been ignored within Snyk's product.
<code>last_resolved</code>	The most recent instance of an issue having been resolved.
<code>reachability</code>	Indicates whether the issue is related to functions that are being called by the application and thus has a greater risk of exploitability.
<code>group_display_name</code>	The display name set for this group.
<code>group_slug</code>	The name of the Group within Snyk.
<code>org_display_name</code>	The display name set for this organization.
<code>org_slug</code>	The name for the Organization within Snyk.
<code>project_name</code>	The name given to this project, when added to Snyk.
<code>project_url</code>	The project URL in Snyk platform
<code>project_is_monitored</code>	Whether this project is currently set to be actively monitored. By the default the API only return monitored projects' issues. To fetch issues of deactivated projects, check the API parameters.
<code>project_type</code>	The scanning method to use for a particular Project, such as Static Application Security Testing (SAST) for scanning using Snyk Code, or Maven for a Maven project using Snyk Open Source. This is part of the configuration for scanning.
<code>project_type_display_name</code>	A display name Snyk assigned to internal project type values.
<code>project_test_frequency</code>	The frequency of testing for a given Project. For example, Daily, Weekly, and so on.
<code>project_origin</code>	The Origin defines the Target ecosystem, such as CLI, GitHub, or Kubernetes. Origins are a property of Targets.
<code>project_target_ref</code>	A reference that differentiates this project, for example, a branch name or version. Projects having the same reference can be grouped based on that reference.
<code>project_target_runtime</code>	The environment in which the Target is executed and run.
<code>project_target_display_name</code>	A display name for a project's target.
<code>project_is_private_target</code>	Whether the target's source is private or publicly reachable
<code>project_target_source_type</code>	The hosting provider of a given target, for example, docker-hub, github, and so on.
<code>project_target_source_type_display_value</code>	A display value that represents the grouping for target sources, for example, Source Control, Container Registry, and so on.

<code>project_target_upstream_url</code>	The URL pointing to a target's upstream source, such as a URL for a GitHub repository.
<code>project_criticalities</code>	A project attribute that indicates business criticality. For example, low, medium, high, critical.
<code>project_lifecycles</code>	A project attribute, for example, production, development, sandbox.
<code>project_environments</code>	A project attribute, for example, frontend, backend, internal, external, mobile, saas, onprem, hosted, distributed.
<code>project_collections</code>	All Project collections to which this project has been added.
<code>project_tags</code>	All tags which have been assigned to this project.
<code>project_owner_email</code>	The email of the user assigned as the owner of this project.

Next

POST Initiate export (Group)



Last updated 3 months ago

## More information



[Snyk privacy policy](#)