The REST API is now versioned. For more information, see "About API versioning."

Dependency submission

Use the REST API to submit dependencies.

About dependency submissions

Note: The ability to use the REST API for dependency submission is currently in public beta and subject to change.

You can use the REST API to submit dependencies for a project. This enables you to add dependencies, such as those resolved when software is compiled or built, to GitHub's dependency graph feature, providing a more complete picture of all of your project's dependencies.

The dependency graph shows any dependencies you submit using the API in addition to any dependencies that are identified from manifest or lock files in the repository (for example, a `package-lock.json` file in a JavaScript project). For more information about viewing the dependency graph, see "Exploring the dependencies of a repository."

Submitted dependencies will receive Dependabot alerts and Dependabot security updates for any known vulnerabilities. You will only get Dependabot alerts for dependencies that are from one of the supported ecosystems for the GitHub Advisory Database. For more information about these ecosystems, see "About the GitHub Advisory database." Submitted dependencies will *not* be surfaced in dependency review or your organization's dependency insights.

You can submit dependencies in the form of a snapshot. A snapshot is a set of dependencies associated with a commit SHA and other metadata, that reflects the current state of your repository for a commit. You can choose to use pre-made actions or create your own actions to submit your dependencies in the required format each time your project is built. For more information, see "Using the Dependency submission API."

You can submit multiple sets of dependencies to be included in your dependency graph. The REST API uses the `job.correlator` property and the `detector.name` category of the snapshot to ensure the latest submissions for each workflow get shown. The `correlator` property itself is the primary field you will use to keep independent submissions distinct. An example `correlator` could be a simple combination of two variables available in actions runs: `<GITHUB_WORKFLOW> <GITHUB_JOB>`.

Create a snapshot of dependencies for a repository

Works with GitHub Apps

Create a new snapshot of a repository's dependencies. You must authenticate using an access token with the `repo` scope to use this endpoint for a repository that the requesting user has access to.

Parameters for "Create a snapshot of dependencies for a repository"

Headers

`accept` string

Setting to `application/vnd.github+json` is recommended.

Path parameters

`owner` string Required

The account owner of the repository. The name is not case sensitive.

`repo` string Required

The name of the repository without the `.git` extension. The name is not case sensitive.

Body parameters

`version` integer Required

The version of the repository snapshot submission.

`job` object Required

▶ Properties of `job`

`sha` string Required

The commit SHA associated with this dependency snapshot. Maximum length: 40 characters.

`ref` string Required

The repository branch that triggered this snapshot.

`detector` object Required

A description of the detector used.

▶ Properties of `detector`

`metadata` object

User-defined metadata to store domain-specific information limited to 8 keys with scalar values.

`manifests` object

A collection of package manifests, which are a collection of related dependencies declared in a file or representing a logical group of dependencies.

▶ Properties of `manifests`

`scanned` string Required

The time at which the snapshot was scanned.

HTTP response status codes for "Create a snapshot of dependencies for a repository"

Status code	Description
`201`	Created

Code samples for "Create a snapshot of dependencies for a repository"

POST /repos/{owner}/{repo}/dependency-graph/snapshots

cURL JavaScript

```
 curl -L \ -X POST \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ http(s)://HOSTNAME/api/v3/repos/OWNER/REPO/dependency-graph/snapshots \ -d '{"version":0,"sha":"ce587453ced02b1526dfb4cb910479d431683101","ref":"refs/heads/main","job":{"correlator":"yourworkflowname_youractionname","id":"yourrunid"},"detector":{"name":"octo-detector","version":"0.0.1","url":"https://github.com/octo-org/octo-repo"},"scanned":"2022-06-14T20:25:00Z","manifests":{"package-lock.json":{"name":"package-lock.json","file":{"source_location":"src/package-lock.json"},"resolved":{"@actions/core":
```

Response

Example response Response schema

```
Status: 201
```

```
 { "id": 12345, "created_at": "2018-05-04T01:14:52Z", "message": "Dependency results for the repo have been successfully updated.", "result": "SUCCESS" }
```

Legal