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

Codespaces repository secrets

Use the REST API to manage secrets for repositories that the user has access to in a codespace.

> **Who can use this feature**
> Users with write access to a repository can manage Codespaces repository secrets.

About Codespaces repository secrets

You can create, list, and delete secrets (such as access tokens for cloud services) for repositories that the user has access to. These secrets are made available to the codespace at runtime. For more information, see "Managing secrets for your codespaces."

List repository secrets

Works with GitHub Apps

Lists all secrets available in a repository without revealing their encrypted values. You must authenticate using an access token with the `repo` scope to use this endpoint. GitHub Apps must have write access to the `codespaces_secrets` repository permission to use this endpoint.

Parameters for "List repository secrets"

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.

Query parameters

`per_page` integer

The number of results per page (max 100).

Default: `30`

---

page integer

Page number of the results to fetch.

Default: `1`

HTTP response status codes for "List repository secrets"

Status code	Description
`200`	OK

Code samples for "List repository secrets"

`GET` `/repos/{owner}/{repo}/codespaces/secrets`

cURL | JavaScript | GitHub CLI

```
curl -L \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ https://api.github.com/repos/OWNER/REPO/codespaces/secrets
```

Response

Example response | Response schema

```
Status: 200
```


```
{ "total_count": 2, "secrets": [ { "name": "GH_TOKEN", "created_at": "2019-08-10T14:59:22Z", "updated_at": "2020-01-10T14:59:22Z", "visibility": "all" }, { "name": "GIST_ID", "created_at": "2020-01-10T10:59:22Z", "updated_at": "2020-01-11T11:59:22Z", "visibility": "all" } ] }
```


Get a repository public key

Gets your public key, which you need to encrypt secrets. You need to encrypt a secret before you can create or update secrets. Anyone with read access to the repository can use this endpoint. If the repository is private you must use an access token with the `repo` scope. GitHub Apps must have write access to the `codespaces_secrets` repository permission to use this endpoint.

Parameters for "Get a repository public key"

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.

HTTP response status codes for "Get a repository public key"

Status code	Description
200	OK

Code samples for "Get a repository public key"

GET /repos/{owner}/{repo}/codespaces/secrets/public-key

cURL | JavaScript | GitHub CLI

```
curl -L \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ https://api.github.com/repos/OWNER/REPO/codespaces/secrets/public-key
```

Response

Example response | Response schema

Status: 200

```
{ "key_id": "012345678912345678", "key": "2Sg8iYjAxxmI2LvUXpJjkYrMxURPc8r+dB7TJyvv1234" }
```

Get a repository secret

Works with GitHub Apps

Gets a single repository secret without revealing its encrypted value. You must authenticate using an access token with the `repo` scope to use this endpoint. GitHub Apps must have write access to the `codespaces_secrets` repository permission to use this endpoint.

Parameters for "Get a repository secret"

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.

`secret_name` string Required

The name of the secret.

HTTP response status codes for "Get a repository secret"

Status code	Description
`200`	OK

Code samples for "Get a repository secret"

`GET` `/repos/{owner}/{repo}/codespaces/secrets/{secret_name}`

cURL | JavaScript | GitHub CLI

```
curl -L \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ https://api.github.com/repos/OWNER/REPO/codespaces/secrets/SECRET_NAME
```

Response

Example response | Response schema

```
Status: 200
```

```
{ "name": "GH_TOKEN", "created_at": "2019-08-10T14:59:22Z", "updated_at": "2020-01-10T14:59:22Z", "visibility": "all" }
```

Create or update a repository secret

Works with GitHub Apps

Creates or updates a repository secret with an encrypted value. Encrypt your secret using LibSodium. For more information, see "Encrypting secrets for the REST API."

You must authenticate using an access token with the `repo` scope to use this endpoint. GitHub Apps must have write access to the `codespaces_secrets` repository permission to use this endpoint.

Parameters for "Create or update a repository secret"

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.

`secret_name` string Required

The name of the secret.

Body parameters

`encrypted_value` string

Value for your secret, encrypted with LibSodium using the public key retrieved from the Get a repository public key endpoint.

`key_id` string

ID of the key you used to encrypt the secret.

HTTP response status codes for "Create or update a repository secret"

Status code	Description
`201`	Response when creating a secret
`204`	Response when updating a secret

Code samples for "Create or update a repository secret"

Example 1: Status Code 201 (application/json)

PUT `/repos/{owner}/{repo}/codespaces/secrets/{secret_name}`

cURL | JavaScript | GitHub CLI

```
curl -L \ -X PUT \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ https://api.github.com/repos/OWNER/REPO/codespaces/secrets/SECRET_NAME \ -d '{"encrypted_value":"c2VjcmV0","key_id":"012345678912345678"}'
```

Response when creating a secret

Example response | Response schema

```
Status: 201
```

Delete a repository secret

Works with GitHub Apps

Deletes a secret in a repository using the secret name. You must authenticate using an access token with the `repo` scope to use this endpoint. GitHub Apps must have write access to the `codespaces_secrets` repository permission to use this endpoint.

Parameters for "Delete a repository secret"

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.

`secret_name` string Required

The name of the secret.

HTTP response status codes for "Delete a repository secret"

Status code	Description
`204`	No Content

Code samples for "Delete a repository secret"

DELETE `/repos/{owner}/{repo}/codespaces/secrets/{secret_name}`

cURL | JavaScript | GitHub CLI

```
curl -L \ -X DELETE \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ https://api.github.com/repos/OWNER/REPO/codespaces/secrets/SECRET_NAME
```

Response

```
Status: 204
```