bitwarden | Help Center Article

SECRETS MANAGER > INTEGRATIONS

GitHub Actions

View in the help center:
https://bitwarden.com/help/github-actions-integration/

GitHub Actions

Bitwarden provides an integration with GitHub Actions to retrieve secrets from Secrets Manager and inject them into GitHub Actions workflows. The integration will inject retrieved secrets as masked environment variables inside an action. To setup the integration:

Save an access token

In this step, we're going to save an access token as a GitHub encrypted secret. Encrypted secrets can be created for an organization, repository, or repository environment and are made available for use in GitHub Actions workflows:

1. In GitHub, navigate to your the repository and select the **Settings** tab.
2. In the Security section of the left navigation, select **Secrets and variables** → **Actions.**
3. Open the **Secrets** tab and select the **New repository secret** button.
4. In another tab, open the Secrets Manager web vault and create an access token.
5. Back in GitHub, give your secret a **Name** like `BW_ACCESS_TOKEN` and paste the access token value from step 4 into the **Secret** input.
6. Select the **Add secret** button.

Add to your workflow file

Next, we're going to add a few steps to your GitHub Actions workflow file.

Get secrets

To get secrets in your workflow, add a step with the following information to your workflow YAML file:

```Bash
- name: Get Secrets
  uses: bitwarden/sm-action@v2
  with:
    access_token: ${{ secrets.BW_ACCESS_TOKEN }}
    base_url: https://vault.bitwarden.com
    secrets: |
      fc3a93f4-2a16-445b-b0c4-aeaf0102f0ff > SECRET_NAME_1
      bdbb16bc-0b9b-472e-99fa-af4101309076 > SECRET_NAME_2
```

Where:

- `${{ secrets.BW_ACCESS_TOKEN }}` references your previously saved repository secret. Change accordingly if you didn't name the secret `BW_ACCESS_TOKEN`.
- `base_url` For self-hosted instances, provide your `https://your.domain.com`. If this optional parameter is provided, the parameters `identity_url` and `api_url` are not required. The GitHub action will use `BASE_URL/identity` and `BASE_URL/api` for the identity and api endpoints.

- `fc3a93f4-2a16-445b-b0c4-aeaf0102f0ff` and `bdbb16bc-0b9b-472e-99fa-af4101309076` reference identifiers for secrets stored in Secrets Manager. The machine account that your access token belongs to **must be able to access these specific secrets**.
- `SECRET_NAME_1` and `SECRET_NAME_2` are the names you'll use to reference the injected secret values in the next step.

Use secrets

Finally, you can complete the pathway by referencing the specified secret names (`SECRET_NAME_1` and `SECRET_NAME_2`) as parameters in a subsequent action, for example:

Bash

```
- name: Use Secret
  run: SQLCMD -S MYSQLSERVER -U "$SECRET_NAME_1" -P "$SECRET_NAME_2"
```

Example workflow

The following example is a Github Actions workflow file using `get secrets`:

Plain Text

```
- name: Get Secrets
        uses: bitwarden/sm-action@v2
        with:
          access_token: ${{ secrets.BW_ACCESS_TOKEN }}
          secrets: |
            fc3a93f4-2a16-445b-b0c4-aeaf0102f0ff > GITHUB_GPG_PRIVATE_KEY
            bdbb16bc-0b9b-472e-99fa-af4101309076 > GITHUB_GPG_PRIVATE_KEY_PASSPHRASE

- name: Import GPG key
        uses: crazy-max/ghaction-import-gpg@v6
        with:
          gpg_private_key: ${{ env.GITHUB_GPG_PRIVATE_KEY }}
          passphrase: ${{ env.GITHUB_GPG_PRIVATE_KEY_PASSPHRASE }}
          git_user_signingkey: true
          git_commit_gpgsign: true
```