

How to use GitHub Actions with security in mind

NDC { Security }



<https://myoctocat.com>

How to use GitHub Actions with security in mind

NDC { Security }

Rob Bos

DevOps Consultant – Xpirit

The Netherlands

<https://devopsjournal.io>

@robbos81



<https://myoctocat.com>

What are GitHub workflows?

Execute one or more **Actions**

Workflows triggered by events:

- Push
- Comment
- Creating an Issue
- Release
- Etc.

What are GitHub Actions?

- Steps in the workflows
- Basis: Run a shell script
- Create your own
- Use an existing one from the **marketplace**



Search or jump to...



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)



[Marketplace](#) / Search results

Types

🔍 Search for apps and actions

Apps

Actions



Actions

An entirely new way to automate your development workflow.

12500

filtered by

Actions



Categories

API management

Chat

Code quality

Code review

Continuous integration

Dependency management

Deployment



Deploy to Cloud Run

By google-github-actions ✓

Use this action to deploy a container in the Google Container Registry to Cloud Run

53 stars



Buildah Build

By redhat-actions ✓

Build a container image, with or without a Dockerfile

36 stars



Amazon ECS "Deploy Task Definition" Action for GitHub Actions

By aws-actions ✓

Registers an Amazon ECS task definition, and deploys it to an ECS service

228 stars



Glo Add Label To Cards

By Axosoft ✓

GitHub action to add a label to Glo Boards cards

3 stars

Workflow example

```
main ▾ dotnetcore-webapp / .github / workflows / dotnetcore.yml
1  name: .NET Core
2
3  on: [push]
4
5  jobs:
6    build-and-deploy:
7      environment: Production
8
9      runs-on: ubuntu-latest
10
11     steps:
12     - uses: actions/checkout@v1
13     - name: Setup .NET Core
14       uses: actions/setup-dotnet@v1
15       with:
16         dotnet-version: 3.0.100
17
18     # dotnet build
19     - name: Build with dotnet
20       run: |
21         dotnet build --configuration Release ./dotnet-core-webapp/dotnetcore-webapp.csproj
22
```



GitHub Actions Security

- **Repository security**
- Runners and security
- Actions and security

- Forking actions
- Keeping up to date

Repository security

- Access to code
- Workflow secrets
- Your code

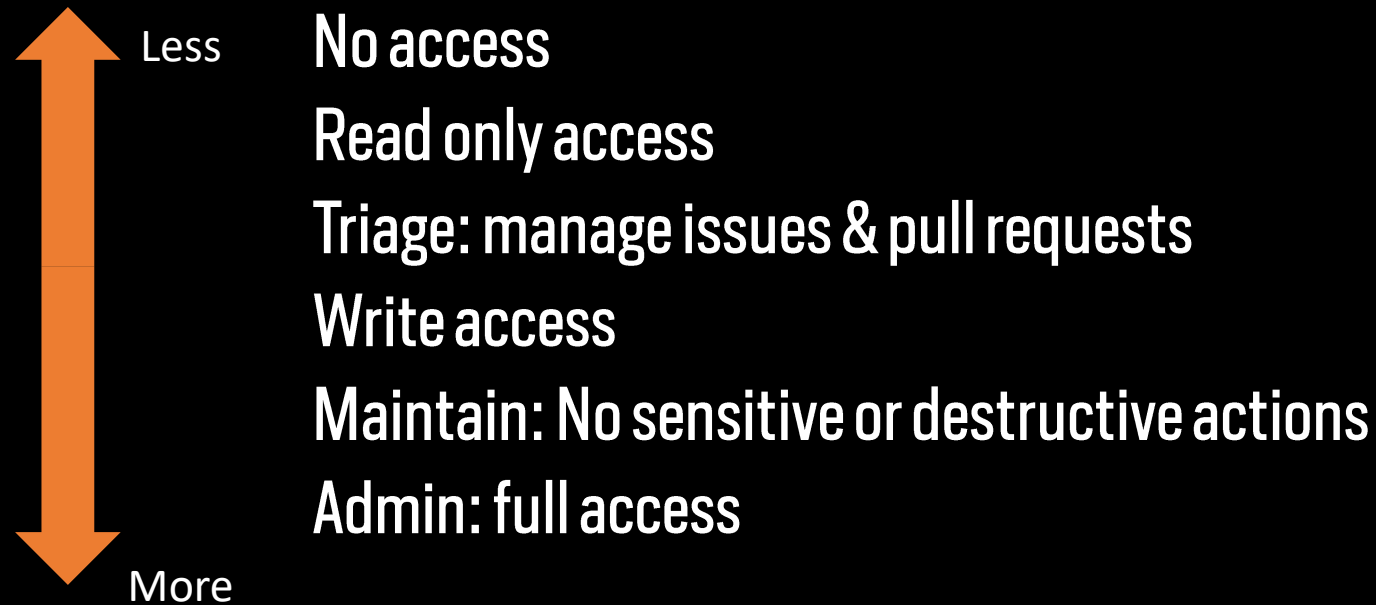
Code – Who has access?

Access levels can be set at:

- Repository
- Organization
- Enterprise

Code – Who has access?

Permission levels



Configuring access

The image shows two overlapping screenshots of the GitHub web interface. The left screenshot displays the 'GlobalDevOpsBootcamp' organization page, with the 'People' tab selected. The right screenshot shows the 'GlobalDevOpsBootcamp / PartsUnlimited-Demo2_2020-Team40' repository page, with the 'Insights' tab selected and the 'People' sub-tab active. Both screenshots have orange boxes highlighting key elements: the organization name, the repository name, the 'People' and 'Insights' tabs, and the 'Insights / People' breadcrumb.

Organization permissions (GlobalDevOpsBootcamp):

- Members: 11
- Outside collaborators
- Pending collaborators
- Pending invitations
- Failed invitations: 1

Repository access configuration (GlobalDevOpsBootcamp / PartsUnlimited-Demo2_2020-Team40):

11 people have access to this repository

User	Role
Magnus Kirø (magnuskiro)	Read
NielsNijveldt	Admin
mericstam (mericstam)	Read
Marcel de Vries (vriesmarcel)	Admin

From the user

The screenshot shows the GitHub profile page for user **rajbos** (Rob Bos) within the organization **GlobalDevOpsBootcamp**. An orange arrow points to the user's profile picture. The user's role is **Owner**, which is highlighted with an orange box. The user has access to 81 repositories and 2 teams, also highlighted with an orange box. The membership is private, two-factor security is enabled, and no SAML identity is linked. Below the profile, there are buttons for "Convert to outside collaborator" and "Remove from organization".

The main content area shows the user's role as **Owner**, with a description: "As an owner, **rajbos** has admin access to all repositories that belong to the GlobalDevOpsBootcamp organization. Manage your owners on the [People page](#)." This section is also highlighted with an orange box. Below this, a search bar allows finding repositories the user has access to. A list of repositories is shown, with the following entries highlighted by orange boxes:

Repository	Role	Action
GlobalDevOpsBootcamp/ PartsUnlimited-Demo2_2020-Team40	Admin on this repository	Manage access
GlobalDevOpsBootcamp/ PartsUnlimited-Demo2020-Team03	Admin on this repository	Manage access
GlobalDevOpsBootcamp/ PartsUnlimited-Demo2_2020-Team41	Admin on this repository	Manage access

Repository security

- Access to code
- Workflow secrets
- Your code

Workflow secrets

@robbos81

Repository secrets

 PUBLISH_PROFILE	Updated on Oct 26, 2019	Update	Remove
 SONAR_TOKEN	Updated on Apr 11, 2020	Update	Remove

```
41
42 # publish to Azure App Service
43 - name: 'Run Azure webapp deploy action using publish profile credentials'
44   uses: azure/webapps-deploy@v2
45   with:
46     app-name: dotnetcorewebapp19 # Replace with your app name
47     publish-profile: ${{ secrets.publish_profile }} # Define secret variable in repository settings as per action documentation
48     package: './dotnetcorewebapp'
```

Workflow secrets

Encrypted client side before reaching GitHub:

- Encrypted with the public key for your org or repo (created and stored by GitHub)
- Used when using the UI
- Encrypt yourself before posting to the REST API

Secrets are **not** shared to forked repositories

Who has access to your secrets?

For creating at **repo** level: Repository Owner access

For creating at **org** level: Admin access to the org

Set an access policy for the secrets:

- All repositories
- Private repositories
- Only selected repositories

Who has access to your secrets?

Encrypted until used, then injected as:

- An environment variable
- Direct input

Will be redacted in logs

Don't use structured data (like json): hard to redact

Who has access to your secrets?

- Actions can do anything with them!
- **Anyone with access to the Action Logs** should be considered to have access to your secrets

```
5 jobs:
6   build-and-deploy:
7
8     runs-on: ubuntu-latest
9
10    steps:
11    - name: Demo secret
12      run: |
13        echo ${ secrets.DEMO_LOG }
14        echo ${ secrets.DEMO_LOG } | sed 's/./& /g'
15
```



build-and-deploy

succeeded 2 minutes ago in 2m 21s

- > ✓ Set up job
- > ✓ Build sonarsource/sonarcloud-github-action@master
- > ✓ Build wei/curl@v1
- ▼ ✓ Demo secret
 - 1 ▶ Run echo ***
 - 6 ***
 - 7 m y - s e c r e t - v a l u e
- > ✓ Run actions/checkout@v1

Repository security

- Access to code
- Workflow secrets
- Your code/repo

Your code

Anything in your repository:

- Workflow files
- Shell scripts
- Your own code
- Dependencies:
 - Packages
 - Containers

Best practices:

- Static code analysis
 - Check your own code!
- Third party dependency scanning
 - 99% of your code, is not yours:
 - Scan for known vulnerabilities
 - Keep your dependencies up to date!

Your code – Best practices

Preventing changes by a single person → Pull Request

Protect your main branch

Branch protection rule:

- 1 approver (4 eyes / 2 person principle)
- Build validation

Your code/repo – trace changes

Who made changes:

- Code: Git commit history
- Everything **around** your code is in the **audit log**

Your code/repo – trace changes (org level)

Audit log:

- Access
- Secrets
- Access Tokens
- OAuth grants
- Enabling features
- Etc.

The screenshot shows the GitHub organization settings page for 'GlobalDevOpsBootcamp'. The 'Settings' tab is highlighted with an orange box. In the left sidebar, the 'Audit log' option is also highlighted with an orange box. The main content area displays the 'Audit log' with a search bar and a list of recent events:

- rajbos – team.add_member**
Added themselves to the [GlobalDevOpsBootcamp/demo-team](#) team
[Netherlands](#) | 14 days ago
- rajbos – team.create**
Created the team [GlobalDevOpsBootcamp/demo-team](#)
[Netherlands](#) | 14 days ago
- MOlausson – org_credential_authorization.grant**
[MOlausson](#) authorized Personal Access Token ***** to access the
[Sweden](#) | on Dec 17, 2020

A red number '23' is visible in the bottom right corner of the screenshot.

Your code/repo – trace changes

Account level:

The screenshot shows the GitHub account settings page for user 'rajbos'. The left sidebar contains a list of settings categories: Profile, Account, Appearance (marked 'New'), Account security, Billing & plans, Security log (highlighted with an orange box), Security & analysis, Emails, Notifications, and Scheduled reminders. The main content area is titled 'Security log' and features a search bar and a 'Filters' dropdown. Below this, the 'Recent events' section lists three actions: 1) 'GitHub System – oauth_authorization.destroy' (Removed authorization for OAuth application was marked as stale (GitHub Co) 9 hours ago), 2) 'rajbos – environment.create_actions_secret' (Created a secret test_env_password for Production 86.93.152.65 | Sint-Michielsgestel, North Brabant, Netherlands | 2 days ago), and 3) 'rajbos – repo.create_actions_secret' (Created a secret for rajbos/dependency-updates 86.93.152.65 | Sint-Michielsgestel, North Brabant, Netherlands | 8 days ago). On the right side, a user profile dropdown menu is open, showing options like 'Set status', 'Your profile', 'Your repositories', 'Your organizations', 'Your enterprises', 'Your projects', 'Your stars', 'Your gists', 'Feature preview', 'Help', 'Settings' (highlighted with an orange box), and 'Sign out'. The top navigation bar includes 'Search or jump to...', 'Pull requests', 'Issues', 'Codespaces', 'Marketplace', and 'Explore'.

GitHub Actions Security

- Repository security
- **Runners and security**
- Actions and security

- Forking actions
- Keeping up to date



Workflow Runners

Actions execute on runners

Self hosted

- Cloud / On premises hosted by yourself
- OS + Tools update = YOUR responsibility
- Enables specific environment setup
- No usage limits

GitHub hosted

- OS + Tools update = GitHub's responsibility
- Per minute rating applies after the free minutes
- Clean execution environment with every run

```
1 name: .NET Core Deploy to IIS
2
3 on:
4   push:
5     branches:
6       - "self-hosted"
7
8 jobs:
9   build-and-deploy:
10
11     runs-on: self-hosted
12
13   steps:
14     - uses: actions/checkout@v1
15     - name: Setup .NET Core
16       uses: actions/setup-dotnet@v1
17       with:
18         dotnet-version: 3.0.100
19
```

```
1 name: .NET Core
2
3 on: [push]
4
5 jobs:
6   build-and-deploy:
7
8     runs-on: ubuntu-latest
9
10   steps:
11     - uses: actions/checkout@v1
12     - name: Setup .NET Core
13       uses: actions/setup-dotnet@v1
14       with:
15         dotnet-version: 3.0.100
16
```

Workflow Runners

Security

- Environment scope
 - Network
 - Shared state between runs
- User: limit its access!

Best practice: Run the action inside of a container

```
jobs:
  my_first_job:
    steps:
      - name: My first step
        uses: docker://gcr.io/cloud-builders/gradle
```



```
jobs:
  test-box:
    runs-on: ubuntu-latest
    container:
      image: azul/zulu-openjdk-alpine:8-jre
    steps:
      - uses: actions/checkout@v2
      - name: What OS is running
        run: uname -a
      - name: What java version do we have
        run: java -version
```

Workflow runners

Best practice: **DO NOT EVER** use self hosted runners for public repositories

Example:

- Your repo
- New fork
- Adds malicious code
- Create pull request to your repo
- Workflow is executed on your self hosted runner?

Persisting data between runs

Run 1:

- Download dependencies
- Build the code
- Somehow overwrite the dependency cache

Run 2:

- Use cached dependencies
- Build the code
- Malicious dependency in build artefact

Solarwind attack:

<https://xpir.it/Solorigate>

Workflow runners – Best practice

Don't share runners (and machines!) between repositories:

- Run 1 can influence Run 2

Risks:

- Malicious programs
- Escaping the runner sandbox
- Exposing access to the (network) environment
- Persisting unwanted or dangerous data



GitHub Actions Security

Repository security

Runners and security

Actions and security

Forking actions

Keeping up to date

Actions

Marketplace or by direct url

Marketplace / Actions / EKS on Fargate

GitHub Action

EKS on Fargate

v0.1.1 Latest version

Use latest version

WIP: Amazon EKS on AWS Fargate

GitHub Actions

Verified creator

GitHub has verified that this action was created by **aws-actions**.

Stars

Star 18

Contributors

aws-actions

EKS on Fargate
Creates and EKS on Fargate cluster

INSTALLATION

Copy and paste the following snippet into your `.yaml` file.

```
- name: EKS on Fargate
  uses: aws-actions/amazon-eks-fargate@v0.1.1
```

Learn more about this action in [aws-actions/amazon-eks-fargate](#)

<https://github.com/aws-actions/amazon-eks-fargate>

Actions and security



Are you running just any
action from the internet?



Scary! Especially in an
enterprise or on local runners

Attack vectors

1. Data Theft
2. Data Integrity Breaches
3. Availability

Protective measures

Manually:

- 1. Check the action repo code before use**
- 2. Check its container images and dependencies before use**

Protective measures

```
uses: shprink/nonharmful-and-must-have-actions@v1
with:
  my-secret: ${{ secrets.MY_SECRET }}
```

<https://github.com/shprink/nonharmful-and-must-have-actions>

If the repo has an **action.yml**, you can use it in your workflow

Protective measures

Only use actions listed in the marketplace?

- There is no real verification process for it 😞

The screenshot shows the GitHub repository page for 'redhat-actions / oc-login'. At the top, there are navigation links for 'Code', 'Issues' (2), 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. On the right, there are buttons for 'Watch' (4), 'Star' (7), and 'Fork' (2). A prominent blue banner with an orange border is highlighted, containing the text: 'Use this GitHub Action with your project' and 'Add this Action to an existing workflow or create a new one.' with a 'View on Marketplace' button. Below the banner, there are repository statistics: 'main' branch, '2 branches', and '4 tags'. A commit history table is visible, showing a commit by 'tetchel' titled 'fix os detection bug' with 40 commits, and two workflow entries: '.github/workflows' (13 days ago) and '__tests__/manifests' (2 months ago). On the right side, there is an 'About' section with a description: 'GitHub Action to log in to an OpenShift cluster and set up a Kubernetes context.' and a list of tags including 'openshift', 'kubernetes', 'k8s', 'oc', 'redhat', 'cloud', and 'action'.

Protective measures

Actions

An entirely new way to automate your development workflow.

45 results for "z" filtered by Actions ×



OWASP ZAP Baseline Scan

By zaproxy

Scans the web application with the OWASP ZAP Baseline Scan
135 stars



Zeebe Action

By jwulf

A GitHub action to interact with Zeebe and Camunda Cloud
6 stars



Verified creator

GitHub has verified that this action was created by **pachyderm**.

[Learn more about verified Actions.](#)



Verified Creator

Verification process:

- GitHub Profile information is present and accurate
- Two factor authentication is on for the organization
- Domain verification through a txt record

See: <https://xpir.it/verified-publisher>

Protective measures

Limiting actions altogether

Actions permissions

- Allow all actions**
Any action can be used, regardless of who authored it or where it is defined.
- Disable Actions**
The Actions tab is hidden and no workflows can run.
- Allow local actions only**
Only actions defined in a repository within rajbos can be used.
- Allow select actions**
Only actions that match specified criteria can be used. [Learn more](#)

Actions permissions

- Allow all actions**
Any action can be used, regardless of who authored it or where it is defined.
- Disable Actions**
The Actions tab is hidden and no workflows can run.
- Allow local actions only**
Only actions defined in a repository within rajbos can be used.
- Allow select actions**
Only actions that match specified criteria can be used. [Learn more about allowing specific actions to run.](#)

- Allow actions created by GitHub**
- Allow Marketplace actions by verified creators**

Allow specified actions

rajbos-actions/*,

Wildcards, tags, and SHAs are allowed. Examples: monalisa/octocat@*, monalisa/octocat@v2, monalisa/*

Protective measures

The screenshot shows a GitHub Actions workflow run that has failed. The workflow is titled "Updating actions with forks (#3) * Update dotnetcore.yml * Update dotnetcore.yml using actions from the rajbos-actions org .NET Core #94". The status is "Startup failure". The failure message, highlighted with an orange box, states: "wei/curl@v1 is not allowed to be used in rajbos/dotnetcore-webapp. Actions in this workflow must be: created by GitHub, within a repository owned by rajbos or match the following: rajbos-actions/*." The error is located in the ".NET Core: .github#L1" step.

rajbos / dotnetcore-webapp

Unwatch 1 Star 0 Fork 110

Code Issues Pull requests Actions Projects Wiki Security 3 Insights

Updating actions with forks (#3) * Update dotnetcore.yml * Update dotnetcore.yml using actions from the rajbos-actions org .NET Core #94

Summary

Jobs

Triggered via push 18 seconds ago	Status	Total duration	Artifacts
rajbos pushed - c64d658 main	Startup failure	-	-

Annotations

1 error

wei/curl@v1 is not allowed to be used in rajbos/dotnetcore-webapp. Actions in this workflow must be: created by GitHub, within a repository owned by rajbos or match the following: rajbos-actions/*.

.NET Core: .github#L1

Protective measures

Pin the action version:

```
uses: gaurav-nelson/github-action-markdown-link-check@v1
```

```
uses: gaurav-nelson/github-action-markdown-link-check@v1.0.1
```

Best practice: Pin the Action's commit SHA:

```
uses: gaurav-nelson/github-action-markdown-link-check@44a942b2f7ed0dc101d556f281e906fb79f1f478
```

Recommendation

- Best practice: Limit to local actions and **fork action repositories**
- Create a separate org to test actions in
 - Enable DevOps teams to own the actions

Actions permissions

- Allow all actions**
Any action can be used, regardless of who authored it or where it is defined.
- Disable Actions**
The Actions tab is hidden and no workflows can run.
- Allow local actions only**
Only actions defined in a repository within rajbos can be used.
- Allow select actions**
Only actions that match specified criteria can be used. [Learn more about allowing specific actions to run.](#)

Workflow attack vectors

- Forks of public repos
- Common fields

Forks of public repos

```
3  on:
4    - push
5    - pull_request
6    - pull_request_target
7
8  jobs:
9    build-and-deploy:
10     environment: PullRequestEnvironment
11
12     runs-on: ubuntu-latest
13
14     steps:
15     - uses: actions/checkout@v1
```

← Safe, runs on merge commit, read only access

← High risks! Runs on the target, has read + write access and can access secrets

<https://xpir.it/gh-pwn-request>

Pull Requests

```
${{ secrets.GITHUB_TOKEN }}
```

Workflow permissions

Choose the default permissions granted to the GITHUB_TOKEN when running workflows in this repository. You can specify more granular permissions in the workflow using YAML. [Learn more.](#)

Read and write permissions

Workflows have read and write permissions in the repository for all scopes.

Read repository contents permission

Workflows have read permissions in the repository for the contents scope only.

Pull Requests

`${{ secrets.GITHUB_TOKEN }}`

```
name: Pull request labeler

on: [ pull_request_target ]

permissions:
  contents: read
  pull-requests: write

jobs:
  triage:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/labeler@v2
        with:
          repo-token: ${{ secrets.GITHUB_TOKEN }}
```


Common fields

```
github.event.issue.title  
github.event.issue.body  
github.event.pull_request.title  
github.event.pull_request.body  
github.event.comment.body  
github.event.review.body  
github.event.review_comment.body  
github.event.pages.*.page_name  
github.event.commits.*.message  
github.event.head_commit.message  
github.event.head_commit.author.email  
github.event.head_commit.author.name  
github.event.commits.*.author.email  
github.event.commits.*.author.name  
github.event.pull_request.head.ref  
github.event.pull_request.head.label  
github.event.pull_request.head.repo.default_branch  
github.head_ref
```

Common fields

```
- name: Check title
  run: |
    title="{{ github.event.issue.title }}"
    if [[ ! $title =~ ^.*:\ .*$ ]]; then
      echo "Bad issue title"
      exit 1
    fi
```

Payload: `a"; echo test`

Remediation

```
- name: print title
```

```
  env:
```

```
    TITLE: ${{ github.event.issue.title }}
```

```
  run: echo '$TITLE'
```

<https://xpir.it/actions-untrusted-input>

GitHub Actions Security

Repository security

Runners and security

Actions and security

Forking actions

Keeping up to date



Forking actions

Pros:

- More secure
- Backup of actions that can be deleted or moved to a different org/repo

Cons:

- More maintenance work
 - Fork needs to be created
 - Kept up to date
- Limits the usage of new actions in your org, as someone create the new action (and by that take responsibility for enabling its use)



GitHub Actions Security

Repository security

Runners and security

Actions and security

Forking actions

Keeping up to date

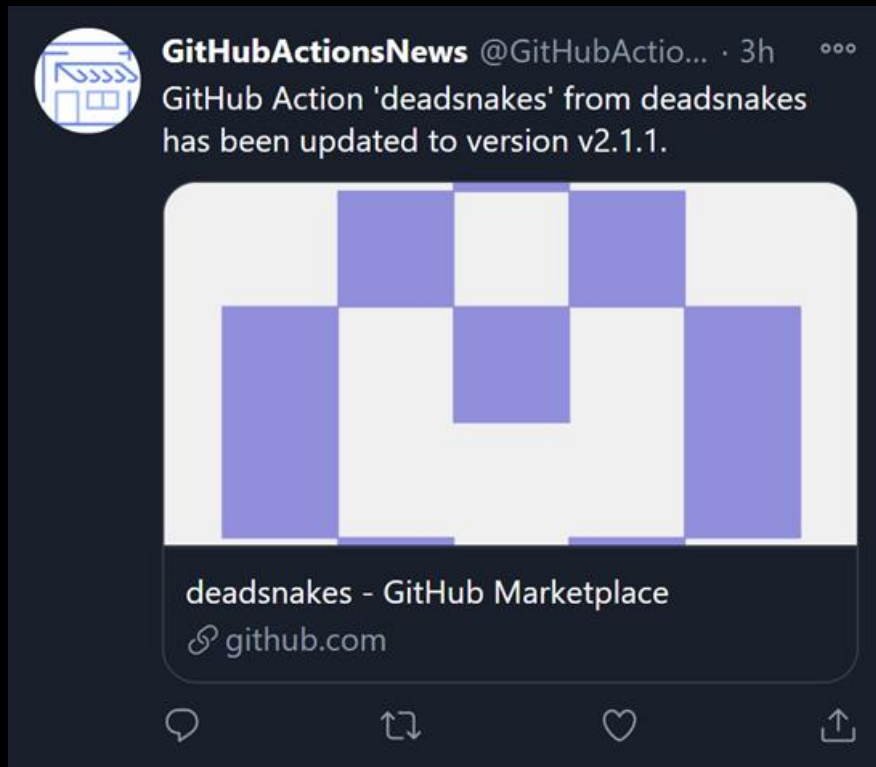
Updates

- Actions are updated regularly
- Wait for a deprecation message?
- **How do you stay up to date?**

- Auto update with a PR?
- Read the changes in the source repo

Staying up to date

Follow [@githubactions](#) on Twitter!



Update action versions

1. Review the Action

Use Actions + Commit SHA + Dependabot

2. Review the Action

Fork the Actions repo, update your forks and use Dependabot

Option 1: Use SHA + Dependabot

Best practice: Pin the Action's commit SHA:

uses: gaurav-nelson/github-action-markdown-link-check@44a942b2f7ed0dc101d556f281e906fb79f1f478

Add `.github/dependabot.yml` to the repo

```
1 #Dependabot will check the dependencies in this repo for updates
2
3 version: 2
4 updates:
5   - package-ecosystem: "github-actions"
6     directory: "/"
7     schedule:
8       # Check for updates to GitHub Actions every weekday
9       interval: "daily"
10
11
12   - package-ecosystem: "nuget"
13     directory: "/"
14     schedule:
15       # Check for updates to on nuget packages every weekday
16       interval: "daily"
```



Use Dependabot

Search or jump to... Pull requests Issues Marketplace Explore

rajbos / dotnetcore-webapp Unwatch 1 Star 1 Fork 114

Code Issues Pull requests 5 Actions Projects Wiki Security 6 Insights

Bump **rajbos-actions/trx-parser** from **v0.0.3** to **v0.0.5** #5 Edit Open with

Open dependabot wants to merge 1 commit into main from dependabot/github_actions/rajbos-actions/trx-parser-v0.0.5

Conversation 1 Commits 1 Checks 3 Files changed 1 +1 -1

Changes from all commits File filter... Jump to... 0 / 1 files viewed Review changes

```
2 .github/workflows/dotnetcore.yml Viewed
```

Line	Change	Code
78		78
79		79 # Using the trx-parser action
80		80 - name: Parse Trx files
81	-	81 uses: rajbos-actions/trx-parser@v0.0.3
81	+	81 uses: rajbos-actions/trx-parser@v0.0.5
82		82 id: trx-parser
83		83 with:
84		84 TRX_PATH: \${{ github.workspace }}\dotnet-core-webapp.webtests\TestResults #This should be the path to your TRX files

Update action versions

1. Review the Action

Use Actions + Commit SHA + Dependabot

2. Review the Action

Fork the **Actions repo**, update your forks and use Dependabot

Keep you forked action up to date

The screenshot shows the GitHub interface for a forked repository. At the top, the repository name is `rajbos-actions / test-repo`, with a note indicating it was forked from `rajbos/test-repo`. Below this, navigation tabs for Code, Pull requests, Actions, Projects, Wiki, and Security are visible. A dropdown menu shows the current branch is `main`. Action buttons include 'Go to file', 'Add file', and 'Code'. A highlighted box contains the message: 'This branch is 2 commits behind rajbos:main.' with links for 'Pull request' and 'Compare'. Below this, a commit history entry shows 'rajbos Initial commit' from 23 hours ago with 1 commit. A file entry for 'README.md' is also shown as an 'Initial commit' from 23 hours ago.

Keep your forked action up to date

Fork a repo and automate it!

<https://github.com/rajbos/github-fork-updater>

Contains:

- Scheduled workflow
- **Creates an issue**
- Review the changes
- Label the issue
- Pull in changes

Creates issues

Search or jump to... / Pulls Issues Codespaces Marketplace Explore

rajbos / github-fork-updater Unwatch 1 Star 0 Fork 0

<> Code Issues 7 Pull requests Actions Projects Wiki

Parent repository for [rajbos/SonarQube-AzureAppService] has updates available #25

Edit New issue

Open github-actions bot opened this issue 22 hours ago · 0 comments

github-actions bot commented 22 hours ago

The parent repository for rajbos/SonarQube-AzureAppService has updates available.

Important!

Click on this [compare link](#) to check the incoming changes before updating the fork.

To update the fork

Add the label update-fork to this issue to update the fork

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

Review before merging

The screenshot shows a GitHub pull request interface for the repository 'rajbos / SonarQube-AzureAppService', which is a fork of 'vanderby/SonarQube-AzureAppService'. The interface includes navigation tabs for Code, Pull requests, Actions, Projects, Security, and Insights. A comparison section titled 'Comparing changes' is highlighted with an orange box, showing the 'base repository' as 'rajbos/SonarQube-AzureAppS...' on the 'base: master' branch and the 'head repository' as 'vanderby/SonarQube-AzureAp...' on the 'compare: master' branch. Below this, it indicates 'Showing 5 changed files with 283 additions and 44 deletions'. The first file shown is '.gitignore', with a diff view showing changes to lines 4 and 5. Line 4 is a new addition: '# Don't include extracted sonarqube folder', and line 5 is another new addition: '+ sonarqube-*/'. The diff view is currently set to 'Unified' mode.

Automation

- Add a label
- Fork gets updated
- Issue gets closed

Parent repository for [rajbos/ParallelTestRunner] has updates available #23

 Closed github-actions bot opened this issue 2 days ago · 2 comments



github-actions bot commented 2 days ago


The parent repository for rajbos/ParallelTestRunner has updates available.

Important!

Click on this [compare link](#) to check the incoming changes before updating the fork.

To update the fork

Add the label `update-fork` to this issue to update the fork

 rajbos added the `update-fork` label now




rajbos commented now

Updating the fork with the incoming changes from the parent repository



rajbos commented now

Fork has been updated

 rajbos closed this now

Pros of forking

- Backup of the action
- Full control over updates
- Pull in updates with validation centrally
- Only allow actions from your actions organization

- Skip commit SHA lookup and updating in every workflow
- Skip adding Dependabot in every repository

<https://xpir.it/actions-best-practices>

GitHub Actions Security

Repository security
Runners and security
Actions and security

Forking actions
Keeping up to date

Best practices summarized

- Treat workflow secrets very carefully: best to think of them as public
- Review actions' source code
- Pin actions to commit SHA
- Don't trust incoming Pull Requests on public repos
- Fork the action repo and limit actions to local actions only
- Have an organization setup to test with
- Keep your forked actions up to date

How to use GitHub Actions with security in mind

NDC { Security }

Rob Bos

DevOps Consultant – Xpirit

The Netherlands

<https://devopsjournal.io>

@robbos81



<https://myoctocat.com>