

Troubleshooting Actions

































Kamran Ayub

Helping you prepare for the GitHub Actions exam

www.linkedin.com/in/kamranayub



Example: Expect to Troubleshoot




























































 kamranayub added 29 commits 2 weeks ago		
 refactor the initial action code	✓	b84fd3a
 try without entrypoint.sh	✗	bc656c
 try ls	✗	ef7d203
 try copy dir	✗	42fd92c
 try moving Docker build context to root	✗	db145a4
 try run opts	✗	1a46a7d
 mount syntax	✗	984b062
 try just cd	✓	d2894fe
 try just js code	✗	9ed9e59
 try slash	✓	d4fcb76
 switch to .net action	✗	95ee3c2
 no sha	✗	8cf0f8f
 no subdir	✗	bc9eab6
 try aspnet and slash	✗	34f752d
 try slash App	✗	d0f3e35
 try workdir	✗	b454a2d
 debug ls -al	✗	37cd4e2
 try root	✓	78a1b8a
 comment for workdir		0215505
 try using sdks	✗	06101d7
 fix syntax error	✓	feb6fad
 formatting	✓	71fba8c
 use AppendAllLinesAsync	✓	5870320
 try local docker script		5ac361f
 mark sh as exec	✓	82f1d65
 update script and it works	✓	bc6c0e7
 try testing a failure	✓	ebb6f85
 test with exit code	✗	4ffe389
 remove test failure	✓	4914d56



**You should expect a lot of
trial and error when testing
GitHub actions**



Example: Expect to Troubleshoot

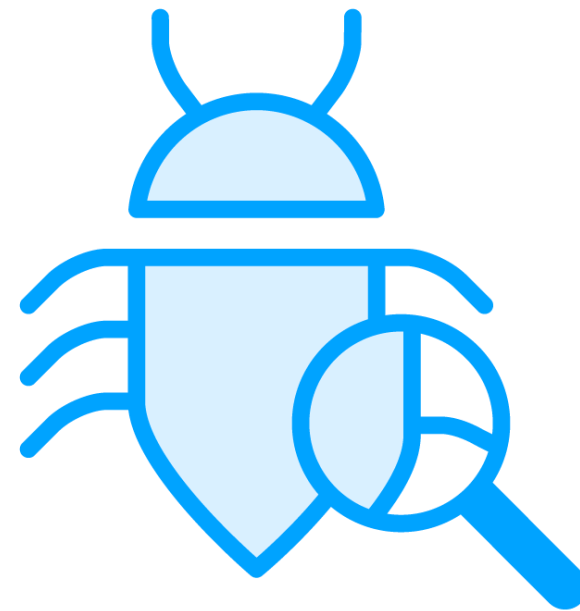
	kamranayub added 29 commits 2 weeks ago	
	 refactor the initial action code	✓ b84fd3a
	 try without entrypoint.sh	✗ bcb656c
	 try ls	✗ ef7d203
	 try copy dir	✗ 42fd92c
	 try moving Docker build context to root	✗ db145a4
	 try run opts	✗ 1a46a7d
	 mount syntax	✗ 984b062
	 try just cd	✓ d2894fe
	 try just js code	✗ 9ed9e59
	 try slash	✓ d4fcb76
	 switch to .net action	✗ 95ee3c2
	 no sha	✗ 8cf0f8f
	 no subdir	✗ bc9eab6
	 try aspnet and slash	✗ 34f752d
	 try slash App	✗ d0f3e35
	 try workdir	✗ b454a2d
	 debug ls -al	✗ 37cd4e2
	 try root	✓ 78a1b8a
	 comment for workdir	0215505
	 try using sdks	✗ 06101d7
	 fix syntax error	✓ feb6fad
	 formatting	✓ 71fba8c
	 use AppendAllLinesAsync	✓ 5870320
	 try local docker script	5ac361f
	 mark sh as exec	✓ 82f1d65
	 update script and it works	✓ bc6c0e7
	 try testing a failure	✓ ebb6f85
	 test with exit code	✗ 4ffe389
	 remove test failure	✓ 4914d56



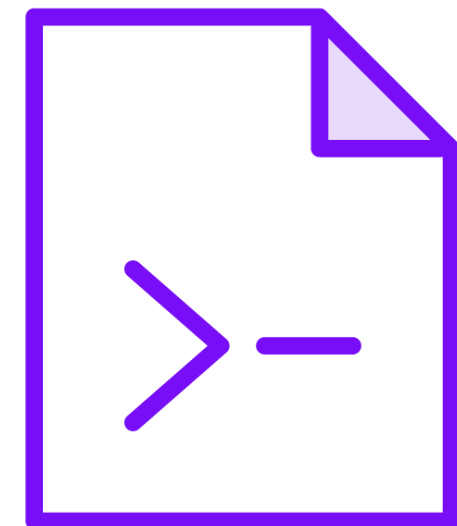
How to Troubleshoot Your Actions



**View workflow run
logs**



Enable debug logging



**Use debug workflow
commands**



Demo: Viewing Workflow Run Logs

← Test Actions

✖ M3: Initial Docker action #38

🔄 Re-run jobs ▾

⋮

🏠 Summary

Jobs

✔ Test Get Employee Action (JavaSc...

✖ Test Get Employee Action (Dock...

✔ Test Send Slack Action (Composite)

Run details

🕒 Usage

📄 Workflow file

Test Get Employee Action (Docker)

failed 2 weeks ago in 4s

🔍 Search logs

🔄

⚙️

> ✔ Set up job

1s

> ✔ Checkout

0s

▾ ✖ Test Get Employee (Docker) Action

0s

1 ▶ Run ./get-employee-docker-action

6 ▼ Building docker image

7 Dockerfile for action: '/home/runner/work/pluralsight-authoring-github-actions-demo-action/pluralsight-authoring-github-actions-demo-action/./get-employee-docker-action/Dockerfile'.

8 /usr/bin/docker build -t 58629f:b8cfd2470b7b59c6517001c543617519 -f "/home/runner/work/pluralsight-authoring-github-actions-demo-action/pluralsight-authoring-github-actions-demo-action/./get-employee-docker-action/Dockerfile" "/home/runner/work/pluralsight-authoring-github-actions-demo-action/pluralsight-authoring-github-actions-demo-action/get-employee-docker-action"

9 #0 building with "default" instance using docker driver

10

11 #1 [internal] load build definition from Dockerfile

12 #1 transferring dockerfile: 784B done

13 #1 DONE 0.0s

14

15 #2 [internal] load metadata for mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff

16 #2 ERROR: mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff: not found

17

18 #3 [internal] load metadata for mcr.microsoft.com/dotnet/sdk:8.0@sha256:35792ea4ad1db051981f62b313f1be3b46b1f45cadbaa3c288cd0d3056eefb83

19 #3 CANCELED

20 -----

21 > [internal] load metadata for mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff:

22 -----

23 Dockerfile:19

24 -----

25 17 |

26 18 | # Build runtime image

27 19 | >>> FROM mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff

28 20 | COPY --from=build-env /out .

29 21 | ENTRYPOINT ["dotnet", "./out/DotNetDockerAction.dll"]

30 -----

31 ERROR: failed to solve: mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff: failed to resolve source metadata for mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff:

mcr.microsoft.com/dotnet/runtime:8.0@sha256:6c4df091e4e531bb93bdbfe7e7f0998e7ced344f54426b7e874116a3dc3233ff: not found

32 Error: Docker build failed with exit code 1

🕒 Print Slack User ID

0s

> ✔ Post Checkout

1s

> ✔ Complete job

0s

🔍 Search logs

🔄

⚙️

Show timestamps

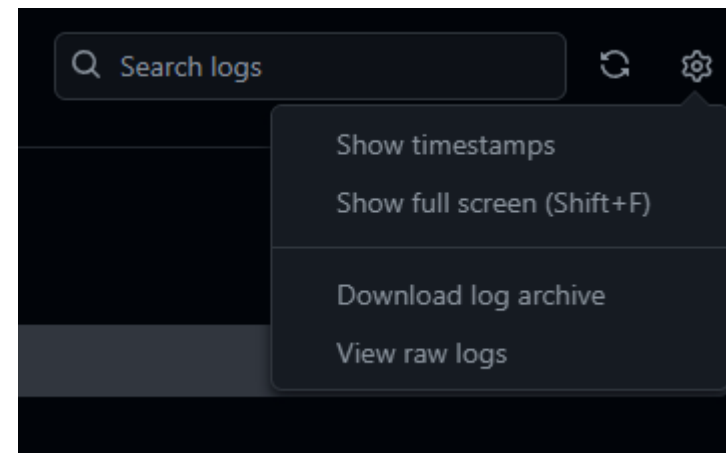
Show full screen (Shift+F)

Download log archive

View raw logs



Demo: Viewing Workflow Run Logs (Cont.)



Demo: Enabling Debug Logging

Re-run single job ✕

A new attempt of this workflow will be started, including Test Get Employee Action (Docker) and dependents:

✕ Test Get Employee Action (Docker)

☐ Enable [debug logging](#) Cancel Re-run jobs

✓ ✕ Test Get Employee (JavaScript) Action

```
1  ##[debug]Evaluating condition for step: 'Test Get Employee (JavaScript) Action'
2  ##[debug]Evaluating: success()
3  ##[debug]Evaluating success:
4  ##[debug]=> true
5  ##[debug]Result: true
6  ##[debug]Starting: Test Get Employee (JavaScript) Action
7  ##[debug]Loading inputs
8  ##[debug]Evaluating: github.actor
```



Demo: Using Debug Workflow Commands

```
23     return core.setFailed(  
24         `No public email found for GitHub user: ${githubUsername}`  
25     );  
26 + } else {  
27 +     core.info(`Email found for GitHub user`);  
28 +     core.debug(`Email: ${email}`);  
29 }  
30  
31 // Slack API: get user by email address  
  
43  
44 // Get the JSON webhook payload for the event that triggered the workflow  
45 const payload = JSON.stringify(github.context.payload, undefined, 2);  
46 + core.debug(`The event payload: ${payload}`);  
47 } catch (error) {  
48     core.setFailed(error.message);  
49 }
```



Demo: Runner Diagnostic Logs

⌵ ⓧ Test Get Employee (JavaScript) Action

```
1  ##[debug]Evaluating condition for step: 'Test Get Employee (JavaScript) Action'
2  ##[debug]Evaluating: success()
3  ##[debug]Evaluating success:
4  ##[debug]=> true
5  ##[debug]Result: true
6  ##[debug]Starting: Test Get Employee (JavaScript) Action
7  ##[debug]Loading inputs
8  ##[debug]Evaluating: github.actor
```

Name	Type
📁 runner-diagnostic-logs	File folder
📄 0_Test Get Employee Action (JavaS...	Text Document
📄 1_Test Send Slack Action (Composi...	Text Document



```
# Add to repository secrets or variables
ACTIONS_STEP_DEBUG=true
ACTIONS_RUNNER_DEBUG=true

# Workflow debug command (Bash)
echo "::debug::A debug log"

# Workflow debug command (GitHub Actions Toolkit)
core.debug("A debug log");

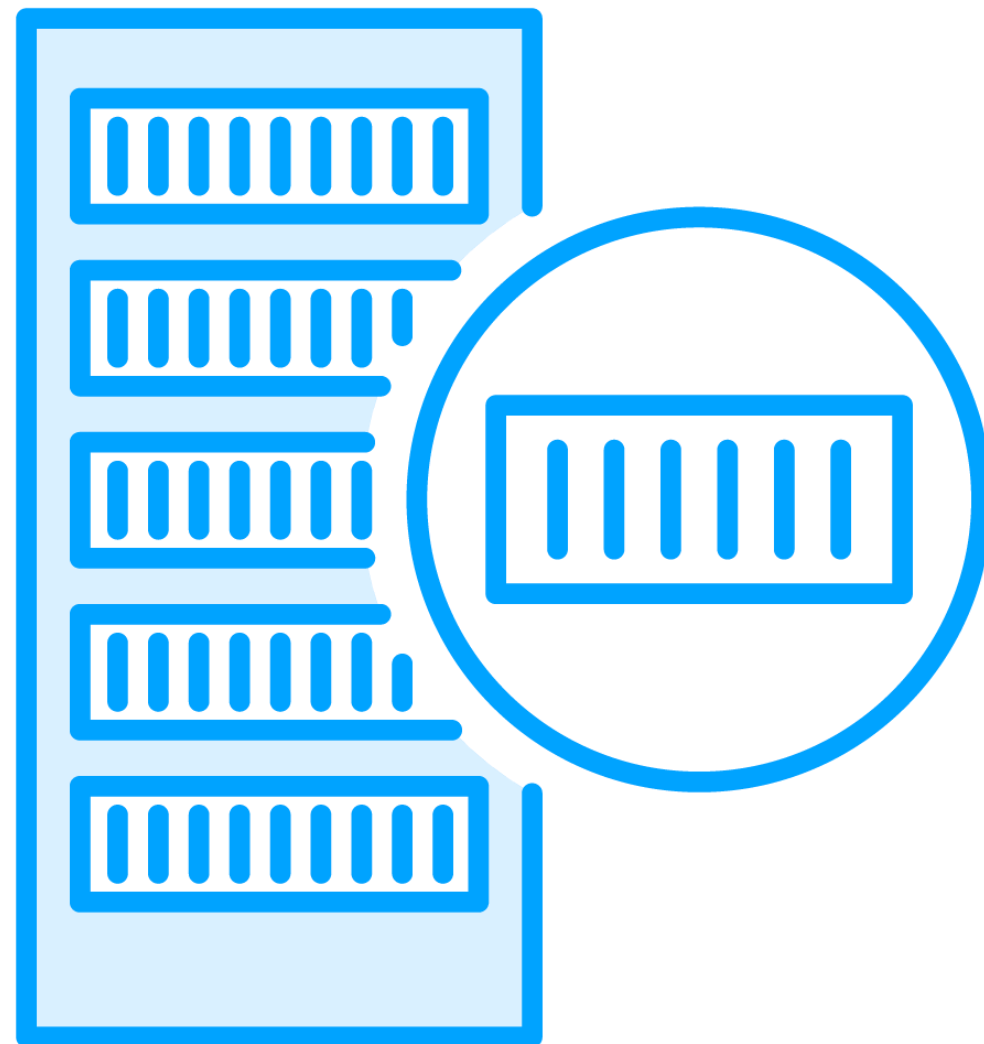
# Workflow debug command (C#)
Console.WriteLine("::debug::A debug log");
```

Enabling Debug Logging for the Runner and Steps

You can add these to the repository variables or secrets. These are set for you when you re-run a workflow with debug logging enabled. Use the debug workflow command to include any verbose output.



Troubleshooting Self-Hosted Runners



Check job logs

- Stored in `_diag` directory
- Filenames start with `Worker_`

Check the runner service

- Using `journalctl`

Check runner connectivity

- Using config script

Make sure Docker is installed

- Required for container actions



References

Troubleshooting Workflows

<https://docs.github.com/en/actions/monitoring-and-troubleshooting-workflows/troubleshooting-workflows/about-troubleshooting-workflows>

Monitoring and Troubleshooting Self-Hosted Runners

<https://docs.github.com/en/actions/hosting-your-own-runners/managing-self-hosted-runners/monitoring-and-troubleshooting-self-hosted-runners>



Troubleshooting JavaScript Actions

Create a test workflow

.github/workflows/test.yml

```
name: Test Actions

on:
  pull_request:
    branches:
      - main
      - m4/demo
  push:
    branches:
      - main

permissions:
  contents: read

jobs:
  test-get-employee-js-action:
    name: Test Get Employee Action (JavaScript)
    runs-on: ubuntu-latest

    steps:
      - name: Checkout
        uses: actions/checkout@v4

      - name: Setup Node.js
        uses: actions/setup-node@v4
        with:
          node-version-file: get-employee-js-action/.nvmrc
          cache-dependency-path: get-employee-js-action
          cache: npm

      - name: Install Dependencies
        run: npm ci
        working-directory: get-employee-js-action

      - name: Test Get Employee (JavaScript) Action
        id: test-employee-js-action
        uses: ./get-employee-js-action
        with:
          github-username: ${ github.actor }
          github-token: ${ secrets.GITHUB_TOKEN }
          slack-token: ${ secrets.SLACK_TOKEN }

      - name: Print Slack User ID
        run: echo "${ steps.test-employee-js-action.outputs.slack-user-id }"
```



Demo: Action Failed Due to Inputs

```
▼ ✓ Test Get Employee (JavaScript) Action  
1 ▼ Run ./get-employee-js-action  
2   with:  
3     github-username: kamranayub  
4     github-token: ***  
5     slack-token: ***
```



Demo: Action Failed Due to API

```

  ▾ ✖ Test Get Employee (JavaScript) Action
    1 ▾ Run ./get-employee-js-action
    2   with:
    3     github-username: kamranayub
    4     github-token: ***
    5     slack-token: ***
    6
    6 Error: An API error occurred: users_not_found

```



Adding Debug Logging to JavaScript Actions

Using the GitHub Actions toolkit

main.js

```
// @ts-check
const core = require("@actions/core");
const github = require("@actions/github");
const { WebClient } = require("@slack/web-api");

run();

async function run() {
  try {
    const githubUsername = core.getInput("github-username");
    const githubToken = core.getInput("github-token");
    const slackToken = core.getInput("slack-token");

    const slackUserId = await lookupSlackUserByGitHubUsername(
      githubUsername,
      githubToken,
      slackToken
    );
    core.setOutput("slack-user-id", slackUserId);
  } catch (error) {
    core.setFailed(error);
  }
}

async function lookupSlackUserByGitHubUsername(
  githubUsername,
  githubToken,
  slackToken
) {
  const octokit = github.getOctokit(githubToken);

  const email = await core.group("Fetch GitHub user info", async () => {
    core.debug(`Looking up email for GitHub user: ${githubUsername}`);

    const {
      data: { email },
    } = await octokit.rest.users.getByUsername({
      username: githubUsername,
    });

    if (!email) {
      throw new Error(
        `No public email found for GitHub user: ${githubUsername}`
      );
    }

    core.debug(`Found email for GitHub user: ${email}`);

    return email;
  });

  return await core.group("Fetch Slack user info", async () => {
    const web = new WebClient(slackToken);

    core.debug(`Looking up Slack user by email: ${email}`);

    const { user: slackUser } = await web.users.lookupByEmail({ email: email });

    if (!slackUser) {
      throw new Error(`No Slack user found with email: ${email}`);
    }

    core.debug(`Found Slack user: ${slackUser.id}`);

    return slackUser.id;
  });
}
```



Demo: Action Debug Logging

```
::group::Fetch GitHub user info
▼ Fetch GitHub user info
  ##[debug]Looking up email for GitHub user: kamranayub
  ##[debug]Found email for GitHub user: kamran.ayub@gmail.com
  ::endgroup::
::group::Fetch Slack user info
▼ Fetch Slack user info
  ##[debug]Looking up Slack user by email: kamran.ayub@gmail.com
  ::endgroup::
Error: Error: An API error occurred: users_not_found
##[debug]Node Action run completed with exit code 1
##[debug]Finishing: Test Get Employee (JavaScript) Action
```



Troubleshooting Docker Actions

Create a test workflow

.github/workflows/test.yml

```
test-get-employee-docker-action:
  name: Test Get Employee Action (Docker)
  runs-on: ubuntu-latest

steps:
  - name: Checkout
    uses: actions/checkout@v4

  - name: Test Get Employee (Docker) Action
    id: test-employee-docker-action
    uses: ./get-employee-docker-action
    with:
      github-username: ${ github.actor }
      github-token: ${ secrets.GITHUB_TOKEN }
      slack-token: ${ secrets.SLACK_TOKEN }

  - name: Print Slack User ID
    run: echo "${ steps.test-employee-docker-action.outputs.slack-user-id }"
```



Demo: Action Failed Due to Entrypoint

```
action/pluralsight-authoring-github-actions-demo-action" /github/w
134 The command could not be loaded, possibly because:
135   * You intended to execute a .NET application:
136     The application 'DotNetDockerAction.dll' does not exist.
137   * You intended to execute a .NET SDK command:
138     No .NET SDKs were found.
139
140 Download a .NET SDK:
141 https://aka.ms/dotnet/download
142
143 Learn about SDK resolution:
144 https://aka.ms/dotnet/sdk-not-found
145
146 Print Stack Trace ID
```



Troubleshooting Docker Actions

Seems to be an issue with the ENTRYPOINT

Dockerfile

```
FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build-env

WORKDIR /App
COPY ./src ./
RUN dotnet restore
RUN dotnet publish -c Release -o out

FROM mcr.microsoft.com/dotnet/aspnet:8.0
WORKDIR /App
COPY --from=build-env /App/out .
ENTRYPOINT ["dotnet", "DotNetDockerAction.dll"]
```



Troubleshooting Docker Actions

Adding a debug log

entrypoint.sh

```
#!/bin/sh

# Show the current working directory
echo "::debug::Current working
directory: $(pwd)"

# Run 'ls -al' and process its output
ls -al | awk '{print "::debug::" $0}'
```

action.yml

```
runs:
  using: docker
  image: Dockerfile

  entrypoint: /entrypoint.sh
```



Demo: Action Container Filesystem

```
-e GITHUB_ACTION -e GITHUB_EVENT_PATH -e GITHUB_ACTION_REPOSITORY -e GITHUB_ACTION_REF -e GITHUB_PATH -e GITHUB_ENV -e GITHUB_STEP_SUMMARY -e GITHUB_STATE -e GITHUB_OUTPUT -e
$ docker: Error response from daemon: failed to create task for container: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: exec:
"/entrypoint.sh": stat ./entrypoint.sh: no such file or directory: unknown.
```



Troubleshooting Docker Actions

Copy over debug entrypoint script

Dockerfile

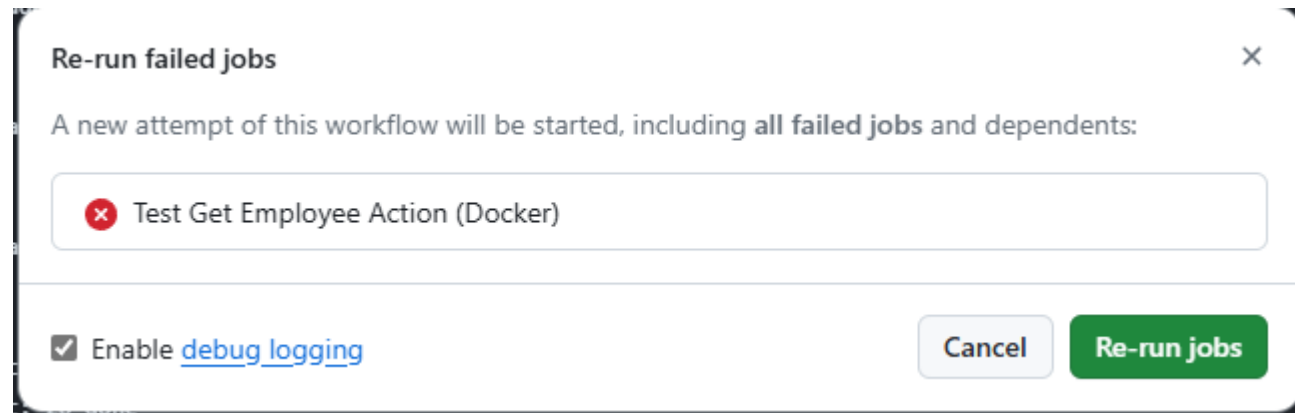
```
FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build-env

WORKDIR /App
COPY ./src ./
COPY entrypoint.sh /entrypoint.sh
RUN dotnet restore
RUN dotnet publish -c Release -o out

FROM mcr.microsoft.com/dotnet/aspnet:8.0
WORKDIR /App
COPY --from=build-env /App/out .
COPY --from=build-env /entrypoint.sh /entrypoint.sh
ENTRYPOINT ["dotnet", "DotNetDockerAction.dll"]
```



Demo: Action Container Filesystem



```
158  ##[debug]Current working directory: /github/workspace
159  ##[debug]total 48
160  ##[debug]drwxr-xr-x 6 1001  127  4096 Aug  6 18:28 .
161  ##[debug]drwxr-xr-x 6 root  root  4096 Aug  6 18:29 ..
162  ##[debug]drwxr-xr-x 8 1001  127  4096 Aug  6 18:28 .git
163  ##[debug]drwxr-xr-x 3 1001  127  4096 Aug  6 18:28 .github
164  ##[debug]-rw-r--r-- 1 1001  127  2066 Aug  6 18:28 .gitignore
165  ##[debug]-rw-r--r-- 1 1001  127 11357 Aug  6 18:28 LICENSE
166  ##[debug]-rw-r--r-- 1 1001  127   451 Aug  6 18:28 README.md
167  ##[debug]-rw-r--r-- 1 1001  127  1183 Aug  6 18:28 action.yml
168  ##[debug]drwxr-xr-x 3 1001  127  4096 Aug  6 18:28 get-employee-docker-action
169  ##[debug]drwxr-xr-x 2 1001  127  4096 Aug  6 18:28 get-employee-js-action
170  ##[debug]Docker Action run completed with exit code 0
171  ##[debug]Finishing: Test Get Employee (Docker) Action
```



Troubleshooting Docker Actions

Copy over debug entrypoint script

Dockerfile

```
FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build-env

WORKDIR /App
COPY ./src ./
COPY entrypoint.sh /entrypoint.sh
RUN dotnet restore
RUN dotnet publish -c Release -o out

FROM mcr.microsoft.com/dotnet/aspnet:8.0
WORKDIR /App
COPY --from=build-env /App/out .
COPY --from=build-env /entrypoint.sh /entrypoint.sh
ENTRYPOINT ["dotnet", "/DotNetDockerAction.dll"]
```



Demo: Action Succeeded

```
144 Found Slack user ID: U02FMV0D57T
```

✓

Print Slack User ID

1

▶ Run echo "U02FMV0D57T"

4

U02FMV0D57T



Troubleshooting Composite Actions

Create a test workflow

.github/workflows/test.yml

```
test-send-message-composite-action:
  name: Test Send Slack Action (Composite)
  runs-on: ubuntu-latest

  steps:
    - name: Checkout
      uses: actions/checkout@v4

    - name: Setup Node.js
      uses: actions/setup-node@v4
      with:
        node-version-file: get-employee-js-action/.nvmrc
        cache-dependency-path: get-employee-js-action
        cache: npm

    - name: Install Dependencies
      run: npm ci
      working-directory: get-employee-js-action

    - name: Test Slack Message (Composite) Action
      id: test-send-action
      uses: ./
      with:
        github-username: ${{ github.actor }}
        github-token: ${{ secrets.GITHUB_TOKEN }}
        slack-token: ${{ secrets.SLACK_TOKEN }}
        slack-bot-token: ${{ secrets.SLACK_BOT_TOKEN }}

    - name: Print Slack Message Time
      run: echo "${{ steps.test-send-action.outputs.time }}"
```

action.yml

```
name: 'Send Status Notification to Slack (Composite)'
description: "Sends the status of the GitHub Action to the user via Slack"

inputs:
  github-token:
    description: 'The GitHub token to authenticate with GitHub API'
    required: true
  github-username:
    description: 'The GitHub username of the employee to look up'
    required: true
  slack-token:
    description: 'The organizational Slack access token to look up users'
    required: true
  slack-bot-token:
    description: 'The organizational Slack access token to send chat messages'
    required: true

runs:
  using: composite
  steps:
    - uses: './get-employee-js-action'
      with:
        github-username: ${{ inputs.github-username }}
        github-token: ${{ inputs.github-token }}
        slack-token: ${{ inputs.slack-token }}

    - name: slack-send
      uses: slackapi/slack-github-action
      with:
        channel-id: ${{ steps.get-employee-js.outputs.slack-user-id }}
        slack-message: "GitHub build result: ${{ job.status }}\n${{
github.event.pull_request.html_url || github.event.head_commit.url }}"
        slack-token: ${{ inputs.slack-bot-token }}
```



Demo: Wrong ref for step

```
1 Error: /home/runner/work/pluralsight-authoring-github-actions-demo-action/pluralsight-authoring-github-actions-demo-action/./action.yml (Line: 29, Col: 11): Expected format {org}/{repo}[/path]@ref. Actual
'slackapi/slack-github-action'
2 Error: System.FormatException: Input string was not in a correct format.
3   at System.Text.ValueStringBuilder.ThrowFormatError()
4   at System.Text.ValueStringBuilder.AppendFormatHelper(IFormatProvider provider, String format, ParamsArray args)
5   at System.String.FormatHelper(IFormatProvider provider, String format, ParamsArray args)
6   at System.String.Format(IFormatProvider provider, String format, Object[] args)
7   at GitHub.Runner.Worker.TemplateTraceWriter.Error(String format, Object[] args)
8   at GitHub.DistributedTask.ObjectTemplating.TemplateContext.Error(Nullable`1 fileId, Nullable`1 line, Nullable`1 column, String message)
9   at GitHub.DistributedTask.ObjectTemplating.TemplateContext.Error(TemplateToken value, String message)
10  at GitHub.DistributedTask.Pipelines.ObjectTemplating.PipelineTemplateConverter.ConvertToStep(TemplateContext context, TemplateToken stepsItem, ReferenceNameBuilder nameBuilder)
11  at GitHub.DistributedTask.Pipelines.ObjectTemplating.PipelineTemplateConverter.ConvertToSteps(TemplateContext context, TemplateToken steps)
12  at GitHub.Runner.Worker.ActionManifestManager.ConvertRuns(IExecutionContext executionContext, TemplateContext templateContext, TemplateToken inputsToken, String fileRelativePath, MappingToken outputs)
13  at GitHub.Runner.Worker.ActionManifestManager.Load(IExecutionContext executionContext, String manifestFile)
14 Error: Failed to load /home/runner/work/pluralsight-authoring-github-actions-demo-action/pluralsight-authoring-github-actions-demo-action/./action.yml
```



Troubleshooting Composite Actions

Fix reference to public action

action.yml

```
- name: slack-send
  uses: slackapi/slack-github-action@v1.26.0
  with:
    channel-id: ${ steps.get-employee-js.outputs.slack-user-id }
    slack-message: "GitHub build result: ${ job.status }\n${
github.event.pull_request.html_url || github.event.head_commit.url }"
    slack-token: ${ inputs.SLACK_BOT_TOKEN }
```



Demo: Input typo

```

✓ ✕ Test Slack Message (Composite) Action

1 Prepare all required actions
2 Getting action download info
3 Download action repository 'slackapi/slack-github-action@v1.26.0' (SHA:70cd7be8e40a46e8b0eced40b0de447bdb42f68e)
4 ▼ Run ./
5   with:
6     github-username: kamranayub
7     github-token: ***
8     slack-token: ***
9     slack-bot-token: ***
10 ▼ Run ./get-employee-js-action
11   with:
12     github-username: kamranayub
13     github-token: ***
14 ▼ Fetch GitHub user info
15 ▶ Fetch Slack user info
16 Error: An API error occurred: not_authed

```

✓ Test Get Employee Action (JavaSc...

✓ Test Get Employee Action (Docker)

✕ Test Send Slack Action (Composi...



Troubleshooting Composite Actions

Fix input typo

action.yml

```
- uses: './get-employee-js-action'  
  with:  
    github-username: ${ inputs.github-username }  
    github-token: ${ inputs.github-token }  
    slack-token: ${ inputs.slack-token }
```



Demo: Wrong way to pass env/input

```
▶ Run ./
▶ Run ./get-employee-js-action
▶ Fetch GitHub user info
▶ Fetch Slack user info
Warning: Unexpected input(s) 'slack-token', valid inputs are ['channel-id', 'slack-message', 'payload', 'payload-file-path', 'payload-file-path-parsed', 'update-ts']
▶ Run slackapi/slack-github-action@v1.26.0
Error: Error: Need to provide at least one botToken or webhookUrl
```



Troubleshooting Composite Actions

Wrong way to pass the token

action.yml

```
- name: slack-send
  uses: slackapi/slack-github-action@v1.26.0
  with:
    channel-id: ${ steps.get-employee-js.outputs.slack-user-id }
    slack-message: "GitHub build result: ${ job.status }\n${
github.event.pull_request.html_url || github.event.head_commit.url }"
    slack-token: ${ inputs.SLACK_BOT_TOKEN }
```



Demo: Slack bot token

Usage

Add this Action as a [step](#) to your project's GitHub Action Workflow file:

```
- name: Post to a Slack channel
  id: slack
  uses: slackapi/slack-github-action@v1.26.0
  with:
    # Slack channel id, channel name, or user id to post message.
    # See also: https://api.slack.com/methods/chat.postMessage#channels
    # You can pass in multiple channels to post to by providing a comma-delimited list of channel IDs.
    channel-id: 'CHANNEL_ID,ANOTHER_CHANNEL_ID'
    # For posting a simple plain text message
    slack-message: "GitHub build result: ${ job.status }}\n${ github.event.pull_request.html_url || g
  env:
    SLACK_BOT_TOKEN: ${ secrets.SLACK_BOT_TOKEN }}
```



Troubleshooting Composite Actions

Wrong way to pass the token

action.yml

```
- name: slack-send
  uses: slackapi/slack-github-action@v1.26.0
  with:
    channel-id: ${ steps.get-employee-js.outputs.slack-user-id }
    slack-message: "GitHub build result: ${ job.status }\n${
github.event.pull_request.html_url || github.event.head_commit.url }"
  env:
    SLACK_BOT_TOKEN: ${ inputs.slack-bot-token }
```



Demo: Slack channel ID

```
✓ x Test Slack Message (Composite) Action

1 Prepare all required actions
2 Getting action download info
3 Download action repository 'slackapi/slack-github-action@v1.26.0' (SHA:70cd7be8e40a46e8b0eced40b0de447bdb42f68e)
4 ▶ Run ./
10 ▶ Run ./get-employee-js-action
15 ▶ Fetch GitHub user info
16 ▶ Fetch Slack user info
17 ▶ Run slackapi/slack-github-action@v1.26.0
24 Channel ID is required to run this action. An empty one has been provided
25 Error: Error: Channel ID is required to run this action. An empty one has been provided
```

Re-run failed jobs

×

A new attempt of this workflow will be started, including all failed jobs and dependents:

x Test Send Slack Action (Composite)

☒ Enable [debug logging](#)

Cancel

Re-run jobs



Demo: Slack channel ID

```
98 ▼ Fetch Slack user info
99   ##[debug]Looking up Slack user by email: kamran@hey.com
100   ##[debug]Found Slack user: U02FMV0D57T
101   ::endgroup::
102   ##[debug]Node Action run completed with exit code 0
103   ##[debug]Set output slack-user-id = U02FMV0D57T
104   ##[debug]Finished: run
105   ##[debug]Evaluating: inputs.slack-bot-token
106   ##[debug]Evaluating Index:
107   ##[debug]..Evaluating inputs:
108   ##[debug]..=> Object
109   ##[debug]..Evaluating String:
110   ##[debug]..=> 'slack-bot-token'
111   ##[debug]=> '***'
112   ##[debug]Result: '***'
113   ##[debug]Evaluating condition for step: 'run'
114   ##[debug]Evaluating: success()
115   ##[debug]Evaluating success:
116   ##[debug]=> true
117   ##[debug]Result: true
118   ##[debug]Starting: run
119   ##[debug>Loading inputs
120   ##[debug]Evaluating: steps.get-employee-js.outputs.slack-user-id
121   ##[debug]Evaluating Index:
122   ##[debug]..Evaluating Index:
123   ##[debug]....Evaluating Index:
124   ##[debug].....Evaluating steps:
125   ##[debug].....=> Object
126   ##[debug].....Evaluating String:
127   ##[debug].....=> 'get-employee-js'
128   ##[debug]....=> null
129   ##[debug]..=> null
130   ##[debug]=> null
131   ##[debug]Result: null
```



Troubleshooting Composite Actions

Missing step ID

action.yml

```
- id: get-employee-js
  uses: './get-employee-js-action'
  with:
    github-username: ${ inputs.github-username }
    github-token: ${ inputs.github-token }
    slack-token: ${ inputs.slack-token }
```



Demo: Action is successful (but still not right)

```
✓ Test Slack Message (Composite) Action

1 Prepare all required actions
2 Getting action download info
3 Download action repository 'slackapi/slack-github-action@v1.26.0' (SHA:70cd7be8e40a46e8b0eced40b0de447bdb42f68e)
4 ▶ Run ./
10 ▶ Run ./get-employee-js-action
15
15 ▶ Fetch GitHub user info
16 ▶ Fetch Slack user info
17 ▼ Run slackapi/slack-github-action@v1.26.0
18   with:
19     channel-id: U02FMV0D57T
20     slack-message: GitHub build result: success
21     https://github.com/kamranayub/pluralsight-authoring-github-actions-demo-action/pull/8
22     payload-file-path-parsed: true
23   env:
24     SLACK_BOT_TOKEN: ***

✓ Print Slack Message Time

1 ▶ Run echo ""
```



Pluralsight Demo GitHub Actions APP 12:01 PM

GitHub build result: success

<https://github.com/kamranayub/pluralsight-authoring-github-actions-demo-action/pull/8>



Troubleshooting Composite Actions

Missing step output

`.github/workflows/test.yml`

```
- name: Print Slack Message Time  
  run: echo "${{ steps.test-send-action.outputs.time }}"
```



Demo: Slack bot token

If you would like to notify the real-time updates on a build status, you can modify the message your build job posted in the subsequent steps. In order to do this, the steps after the first message posting can have `update-ts: ${steps.slack.outputs.ts }` in their settings. With this, the step updates the already posted channel message instead of posting a new one.



Troubleshooting Composite Actions

Missing step output

action.yml

```
outputs:
  time:
    description: 'The time the message was sent'
    value: ${ steps.slack-send.outputs.ts }

runs:
  using: composite
  steps:

  - id: slack-send
```



Demo: Action is successful

✓ Test Slack Message (Composite) Action


```
1 Prepare all required actions
2 Getting action download info
3 Download action repository 'slackapi/slack-github-action@v1.26.0' (SHA:70cd7be8e40a46e8b0eced40b0de447bdb42f68e)
4 ▶ Run ./
10 ▶ Run ./get-employee-js-action
15
15 ▶ Fetch GitHub user info
16 ▶ Fetch Slack user info
17 ▶ Run slackapi/slack-github-action@v1.26.0
```

✓ Print Slack Message Time

```
1 ▶ Run echo "1723050925.702359"
4 1723050925.702359
```





Demo: GitHub Actions extension




GitHub Actions

v0.26.3

GitHub  github.com |  2,471,067 | ★★★★★ (44)

GitHub Actions workflows and runs for github.com hosted repositories in VS Code

[Disable](#) | [Uninstall](#) | 

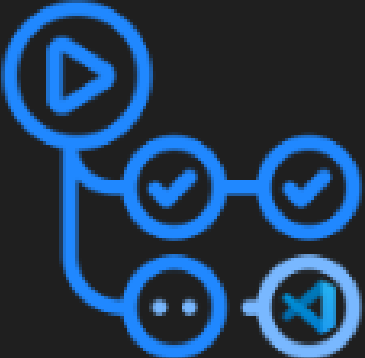
```
- name: Test Get Employee (JavaScript) Action
  id: test-employ
  uses: ./get-emp
  with:
    github-username:
    github-token:
    slack-token: ${{ secrets.SLACK_TOKEN }}
```

Context access might be invalid: SLACK_TOKEN

Contains the names and values of secrets that are available to a workflow run. For more information, see [secrets context](#).


[View Problem \(Alt+F8\)](#) [Quick Fix... \(Ctrl+.\)](#)

You, 2 days ago • Add test workflow and nvmrc




GitHub Actions

v3.0.1

Mathieu Dutour |  129,341 | ★★★★★ (3)

YAML schema validation and auto-completion for GitHub Actions.

[Disable](#) | [Uninstall](#) | 

```
using: composite
st
-
  with:
```

Property withs is not allowed. yaml-schema: GitHub Action

"withs": Unknown word. cSpell

[View Problem \(Alt+F8\)](#) [Quick Fix... \(Ctrl+.\)](#)

