This document includes supplementary information for the user study conducted in the following paper: “Action Required: A Mixed-Methods Study of Security Practices in GitHub Actions.”

Table of Contents

Recruitment Email

This section presents the email sent during the recruitment phase of our user study.

Subject: A Survey on the Use of GitHub Actions in Public Repositories

Dear Developer Using GitHub Actions,

We are researchers from Waseda University, NTT, and NTT DOCOMO BUSINESS in Japan, studying how developers use GitHub Actions in public repositories. Your contact information was obtained solely through publicly available sources. Specifically, we analyzed the commit histories of public repositories to identify GitHub accounts that contribute to GitHub Actions, and we are reaching out using the email address listed on your GitHub profile. Please note that we did not use any email addresses from commit metadata.

This survey asks specific questions about your use of GitHub Actions, such as your goals, configuration strategies, and typical practices. The survey contains up to 56 questions and is designed to be completed within 20 minutes.

We greatly appreciate developers like you supporting this research!

You are eligible for the study if you:

1. Are at least 18 years old
2. Are comfortable completing the study in English.
3. Have experience using GitHub Actions in public repositories - specifically, if you have contributed to YAML files located in the '.github/workflows' directory.

While there is no monetary compensation for participating, your responses will provide valuable insights into how GitHub Actions are used in practice. Our goal is to contribute to the developer community through this research. The cooperation of each and every one of you will lead to a more realistic understanding and better support in the future.

The results of the study may be published in academic papers or presented at conferences. We will ensure that no individual or affiliation is identifiable in any publication. If you are interested, we can share a summary of the aggregated findings with you once the study is complete.

If you're eligible and would like to participate, please use the following link (click or paste into your browser):
<link to the survey>

Thank you very much for your time and consideration.

Best regards,
On behalf of the project team,
<Contact information of authors>

Description of Security Practices

This section presents the full text of the descriptions for each security practice, which were developed in Section VI-A (Survey Design) and Appendix F (Summarizing Documentation of Security Practices) of the main paper. They were shown to participants before the corresponding survey questions.

P1: Using CODEOWNERS to monitor changes.

You can use the CODEOWNERS feature to control how changes are made to your workflow files. A CODEOWNERS file defines individuals or teams responsible for code in a repository. To use a CODEOWNERS file, create a new file called CODEOWNERS in the .github/, root, or docs/ directory of the repository, in the branch where you'd like to add the code owners. It uses a pattern that follows most of the same rules used in gitignore files. The pattern is followed by one or more GitHub usernames or team names using the standard @username or @org/team-name format. This ensures that when someone opens a pull request that modifies owned code, the appropriate reviewers are automatically requested, allowing you to control how changes are made. For example, if all your workflow files are stored in .github/workflows, you can add this directory to the code owners list, so that any proposed changes to these files will first require approval from a designated reviewer.

P2: Good practices for mitigating script injection attacks.

There are a number of different approaches available to help you mitigate the risk of script injection:

- Using an action instead of an inline script (recommended): The recommended approach is to create a JavaScript action that processes the context value as an argument. This approach is not vulnerable to the injection attack, since the context value is not used to generate a shell script, but is instead passed to the action as an argument.
- Using an intermediate environment variable: For inline scripts, the preferred approach to handling untrusted input is to set the value of the expression to an intermediate environment variable, which is stored in memory and used as a variable, and doesn't interact with the script generation process.

Note: A script injection attack can occur when an attacker adds malicious commands and scripts to a context, and the workflow interprets those strings as code which is then executed on the runner.

P3: Using OpenSSF Scorecard to secure workflows and repository.

Scorecard is an automated security tool that flags risky supply chain practices. Once configured using the Scorecard action and workflow template, it runs automatically on repository changes and alerts developers through the built-in code scanning experience. Checks include script injection attacks, token permissions, and pinned actions.

Note: Scorecard is an automated tool that assesses a number of important heuristics associated with software security and assigns each check a score of 0–10. It helps open source maintainers improve their security best practices and helps consumers judge whether their dependencies are safe. The Scorecard action is available on the GitHub Marketplace, and a workflow template is included in GitHub's official starter workflows.

P4-1: Pin third-party actions to a full length commit SHA

Pinning an action to a full length commit SHA is currently the only way to use an action as an immutable release. Pinning to a particular SHA helps mitigate the risk of a bad actor adding a backdoor to the action's repository, as they would need to generate a SHA-1 collision for a valid Git object payload. When selecting a SHA, you should verify it is from the action's repository and not a repository fork.

P4-2: Pin third-party actions to a tag only if you trust the creator

Although pinning to a commit SHA is the most secure option, specifying a tag is more convenient and is widely used. If you'd like to specify a tag, then be sure that you trust the action's creators. The 'Verified creator' badge on GitHub Marketplace is a useful signal, as it indicates that the action was written by a team whose identity has been verified by GitHub. Note that there is risk to this approach even if you trust the author, because a tag can be moved or deleted if a bad actor gains access to the repository storing the action.

P5: Using Dependabot version updates to keep actions up to date

You can use Dependabot to ensure that references to actions and reusable workflows used in your repository are kept up to date. Actions are often updated with bug fixes and new features to make automated processes faster, safer, and more reliable. Dependabot takes the effort out of maintaining your dependencies as it does this automatically for you.

Configure Dependabot by creating a `dependabot.yml` file in the `.github` directory. Specify "`github-actions`" as the `package-ecosystem`, set the `directory` to `/`, and define a `schedule.interval`. Dependabot will raise pull requests for version updates for any outdated actions it finds.

Survey Questions for User Study

This section presents the full text of the questionnaire used in our user study. Note that it includes some supplemental questions that are not described in the main text.

A Survey on the Use of GitHub Actions in Public Repositories

Purpose of this survey:

- The purpose of this questionnaire is to understand how developers use GitHub Actions in practice and to share the insights gained with the broader developer community.
- The contents of the answers for the questionnaires may be published in academic publications. We will make sure that individuals/organizations will not be identified if the results of the questionnaire are published in academic publications.

Survey Eligibility:

- This survey is intended for developers who have experience using GitHub Actions in public repositories. Specifically, it targets those who have contributed to YAML files located under the .github/workflows directory (hereafter referred to as ”workflow files”).

Data handling:

- This survey is conducted anonymously, and no personally identifiable information/organization-identifiable information will be collected.
- Your responses will only be used for the purposes of this research.
- The contents of the answers to the questionnaire will be treated as confidential and strictly managed to ensure that they are not lost or leaked.

Compensation:

- There is no monetary compensation for participating in this survey.
- If you’ re interested, a summary of the aggregated results will be shared with you at a later date.

Other:

- Please note that all questions in this questionnaire are optional. If you do not want to answer a question due to your position or affiliation, please leave it blank/unselected. Also, you can stop answering midway through the questionnaire.

Number of questions: 56 (MAX)
Duration: 20 minutes
Contacts: For additional questions about this research, you may contact:
<Contact information of authors>

Questions about GitHub Repository.

Introduction.

The following questions are about a public repository you are involved in developing. Please answer based on one public repository where you have configured GitHub Actions workflow files (e.g., created, modified, or deleted). If you have configured workflow files in multiple public repositories, please choose the one you are most actively involved in. We recommend that you open the repository's page in a different browser or tab while answering the following questions.

R1. Which programming languages are used in the repository you are involved in? Please refer to the "Languages" section on the repository page and select all that apply. If "Other" is included in the repository's language list, you do not need to enter "Other" in the "Not listed above" field.

1. Python
2. JavaScript
3. TypeScript
4. Java
5. C++
6. Go
7. PHP
8. C#
9. Jupyter Notebook
10. C
11. Shell
12. Ruby
13. Rust
14. Objective-C
15. Swift

16. Kotlin
17. Lua
18. R
19. Tex
20. Haskell
21. Dockerfile
22. Perl
23. HTML
24. CSS
25. Not listed above (please specify)

R2. How many contributors are there in the repository you are involved in? Please refer to the “Contributors” section on the repository page and select the appropriate option. If there are no contributors (e.g., the project is developed individually), please select 0.

1. 0 (No contributors)
2. 1-4
3. 5-9
4. 10-49
5. 50-99
6. 100-499
7. 500-999
8. 1,000-1,999
9. 2,000-4,999
10. 5,000-9,999
11. 10,000 or over

R3. How many stars does the repository you are involved in have? Please refer to the number of stars on the repository page and select the appropriate option.

1. 0
2. 1-9
3. 10-49
4. 50-99
5. 100-499
6. 500-999
7. 1,000-4,999
8. 5,000-9,999
9. 100,000 or over

10. 10,000-49,999
11. 50,000-99,999
12. 100,000 or over

R4. How many commits does the repository you are involved in have? Please refer to the number of commits shown on the repository page and select the appropriate option.

1. 0
2. 1-9
3. 10-49
4. 50-99
5. 100-499
6. 500-999
7. 1,000-4,999
8. 5,000-9,999
9. 100,000 or over
10. 10,000-49,999
11. 50,000-99,999
12. 100,000 or over

R5. When was the most recent commit in the repository you are involved in? Please refer to the latest commit information on the repository page and select the appropriate option.

1. Within the past 24 hours
2. Within the past 7 days
3. Within the past 30 days
4. Within the past 3 months
5. Within the past 6 months
6. Within the past year
7. Within the past 5 years
8. More than 5 years ago

R6. Is the repository you are involved in owned by a user account or an organization account?

1. Organization account
2. User account

R7. This is a question for those who selected “Organization account” in the previous question (R6). What is your assigned role in the repository you are involved in?

1. Admin (You have administrative permissions on the repository. This includes cases where you are an Organization Owner with admin access to the repository.)
2. Organization Member (You are a member of the organization with a role such as Maintainer, Write, or Triage access.)
3. Outside Collaborator (You have been invited to collaborate on the repository but are not a member of the organization.)
4. Contributor (You are not invited to the repository but have contributed through issues, pull requests, or other activities.)
5. Not sure

R8. This is a question for those who selected “User account in the previous question (R6). What is your assigned role in the repository you are involved in?

1. Owner (You are the owner of the repository.)
2. Collaborator (You have been invited as a collaborator on the repository.)
3. Contributor (You have not been invited to the repository but have contributed through issues, pull requests, etc.)
4. Not sure

R9. This is a question for those who selected “Admin” in the question R7 or “Owner” in the question R8. What is the current Workflow permissions setting for the repository you are involved in?

1. Read and write permissions
2. Read repository contents and packages permissions
3. Read and write permissions + Allow GitHub Actions to create and approve pull requests
4. Read repository contents and packages permissions + Allow GitHub Actions to create and approve pull requests
5. Prefer not to answer / Not sure

R10. This is a question for those who answered that "Allow GitHub Actions to create and approve pull requests" is enabled in the previous question (R9). Why do you allow GitHub Actions to create and approve pull requests in your repository? Please select the one that most closely applies.

1. To support automation and improve development efficiency (e.g., Enabled as part of automation to ensure more stable and efficient development.)
2. Because rapid response is required in the development process (e.g., Frequent updates or tight deployment timelines make manual handling difficult.)
3. It is required due to the nature of the system or development workflow (e.g., The process is designed with automated PR creation and approval as a fundamental component.)
4. It's not strictly necessary but was enabled due to practical constraints (e.g., In cases where PR approvals are mandatory in personal development, GitHub's specifications prevent self-approval, so it's necessary to have GitHub Actions approve the PR.)
5. No particular reason (e.g., Enabled without deliberate decision-making.)
6. I don't know, someone else configured it (e.g., The setting was made by another team member, and the reason is unclear.)
7. I don't remember (e.g., Set up in the past, but the reason is forgotten.)
8. Prefer not to answer/Not sure
9. Other. (Please specify)

Questions about GitHub Actions.

Introduction.

The following questions are about Github Actions used in a public repository you are involved in developing. Please answer based on one public repository where you have configured GitHub Actions workflow files (e.g., created, modified, or deleted). If you have configured workflow files in multiple public repositories, please choose the one you are most actively involved in. We recommend that you open the repository's page in a different browser or tab while answering the following questions.

GA1. What kinds of tasks are automated within Github Actions workflow jobs in the repository you are involved in? Please select all that apply. [Multiple options can be selected.]

1. Benchmarking (e.g., running and recording benchmarks)
2. Build (e.g., building or packaging software)
3. Code Review Support (e.g., posting automated comments, assisting with code reviews)
4. Code Scanning (e.g., scanning for vulnerabilities in code)
5. Container CI (e.g., building and testing Docker containers, pushing to registries)
6. Dependency Management (e.g., updating or monitoring dependencies)

7. Deployment (e.g., deploying to production or staging environments)
8. Documentation (e.g., auto-generating or publishing documentation)
9. Formatting (e.g., automatic code formatting, running formatters)
10. Linting (e.g., syntax checks, style violations, type checks)
11. Localization (e.g., managing or validating translation files)
12. Mobile CI (e.g., building and testing mobile applications)
13. Monitoring (e.g., tracking performance or availability)
14. Project Management (e.g., managing issues or pull requests)
15. Publishing (e.g., publishing packages to npm, PyPI, etc.)
16. Reporting (e.g., generating test results or build status reports)
17. Security (e.g., detecting leaked secrets, checking security settings or policies)
18. Scheduled Jobs (e.g., tasks triggered on a cron schedule)
19. Testing (e.g., running unit or integration tests)
20. Other. (Please specify)

GA2. When configuring GitHub Actions, which sources of information do you typically refer to? Please select all that apply. [Multiple options can be selected.]

1. GitHub official documentation
2. Web articles published by experts (e.g., blogs)
3. Developer forums/Q&A websites (e.g., Reddit, Stack Overflow)
4. AI-based tools: LLMs (e.g., ChatGPT), Code Agents (e.g., GitHub Copilot)
5. Configuration examples from your own or your organization's repositories
6. Configuration examples from repositories outside of yourself or your organization
7. Other. (Please specify)

GA3. If GitHub Actions becomes temporarily unavailable, how serious of an issue would that be for your repository?

1. Very serious (Would cause significant disruption or loss. e.g., release delays, complete halt in development or operations, impacts on external teams)
2. Somewhat serious (Causes inconvenience but is manageable. e.g., some delays, but can be worked around or tolerated for a short time)
3. Not very serious (Minimal impact. e.g., CI is only supplementary; development can continue without it)
4. Not a problem at all (No impact on work or activities. e.g., the project does not rely on CI/CD availability)
5. Not sure

GA4. In the repository you are involved in, do the GitHub Actions workflows use any sensitive information? Examples of sensitive information include: GitHub credentials (e.g., access tokens), Credentials for external services (e.g., access tokens, API keys), Cloud platform credentials (e.g., usernames/passwords, SSH keys, API keys), Environment configuration values (e.g., database information), User data or personal information, Software license keys.

1. Yes
2. No
3. Not sure

GA5. In the repository you're involved in, do any of the workflows use actions? Note: An action is an individual task that runs as a step within a job in a GitHub Actions workflow. Actions are referenced using the uses keyword in the `jobs.<job_id>.steps` section. You can use actions from the same repository, a public repository, or a Docker container image.

1. Yes
2. No
3. Not sure

Questions about Security Practices

Introduction.

The following section describes features and settings that are recommended to enhance the security of GitHub Actions (security practices). Please read the descriptions carefully before answering the subsequent questions. Note: When answering these questions, please answer based on one public repository where you have configured GitHub Actions workflow files (e.g., created, modified, or deleted). If you have configured workflow files in multiple public repositories, please choose the one you are most actively involved in. We recommend that you open the repository's page in a different browser or tab while answering these questions. However, please answer based solely on the content of the repository and your own experience — do not use web searches or external sources.
[At this point, we presented participants with a description of security practice presented in the previous section (Description of Security Practices) and then asked the questions SP1–SP4. This procedure was repeated six times, once for each of the following practices: P1, P2, P3, P4-1, P4-2, and P5.]

SP1: Do you currently implement this security practice in the repository you are involved in?

1. Yes, I currently implement this practice.
2. No, I have never implemented this practice.

3. No, I used to implement this practice but no longer do.
4. No, but I achieve the same goal using a different approach. (Please describe your approach),
5. Not sure whether we use this practice or not.,
6. I don't understand what this practice means.

SP2. This is a question for those who answered "No" in the previous question. What is the main reason you are not implementing this security practice? Please select the one that most closely applies.

1. I was not aware of this security practice.
2. The configuration required to implement this security practice is difficult.
3. Implementing this practice would increase maintenance or operational costs.
4. Implementing this practice requires a significant learning effort.
5. I'm concerned that this practice could negatively impact development.
6. I don't understand the risks this practice is meant to address or the benefits it provides.
7. I tried to implement it but gave up due to technical issues (e.g., errors).
8. I tried to implement it but couldn't get approval or alignment from team members or project leads.
9. This practice is unnecessary or overly strict for the repository I'm working on.
10. Other. (Please specify)

SP3. This is a question for those who answered "I was not aware of this security practice." in the previous question. Are you willing to implement this security practice in the repository you are involved in?

1. Yes
2. No
3. Not sure

SP4. This is a question for those who answered "No" in the previous question. What is the main reason you are not willing to implement this security practice? Please select the one that most closely applies.

1. The configuration required to implement this security practice is difficult.
2. Implementing this practice would increase maintenance or operational costs.
3. Implementing this practice requires a significant learning effort.
4. I'm concerned that this practice could negatively impact development.
5. I don't understand the risks this practice is meant to address or the benefits it provides.
6. I'm concerned that I might run into technical issues (e.g., errors) if I try to implement it.

7. I'm concerned that I might not be able to get approval or alignment from team members or project leads if I try to implement it.
8. This practice is unnecessary or overly strict for the repository I'm working on.
9. Other. (Please specify)

SP5. For each of the following security practices, how important do you think it is for the repository you are involved in? Please rate each practice on a 6-point scale. [6-point Likert scale: "Extremely important", "Very important", "Somewhat important", "Not very important", "Not important", "Not important at all"]

SP6. For each of the following security practices, how easy or difficult do you think it is to implement in the repository you are involved in? Please rate each practice on a 6-point scale. [6-point Likert scale: "Very easy", "Easy", "Somewhat easy", "Somewhat difficult", "Difficult", "Very difficult"]

SP7. Regarding GitHub Actions security practices, please select up to three reasons that apply to you - either why you are currently implementing them or why you might consider doing so in the future. If you're not currently implementing these practices, please respond based on why you would choose to do so in the future. [Up to three options can be selected.]

1. It naturally became part of our development process.
2. I recognize that the repository requires strong security (e.g., due to expectations from users or stakeholders).
3. We adopted best practices that were already implemented in other projects.
4. A team member suggested implementing them.
5. The project involves business or commercial services, and security was deemed necessary.
6. We were required to follow internal company policies or guidelines.
7. These practices were recommended by external guidelines or industry standards.
8. We wanted to improve the public trust or reputation of the project.
9. I saw security incidents related to CI/CD on social media or in the news.
10. The repository had previously been flagged for vulnerabilities or experienced an attack.
11. External parties (e.g., organizations or users) suggested or requested it.
12. Other. (Please specify)

SP8. When implementing GitHub Actions security practices (or if you were to implement them in the future), please select up to three aspects that you consider important. If you're not currently implementing these practices, please answer based on what you would prioritize if you were to do so in the future. [Up to three options can be selected.]

1. Easy to set up and quick to adopt

2. Low maintenance and operational overhead
3. Low learning curve; easy to understand and use
4. Does not interfere with the existing development workflow or process
5. Clearly effective in mitigating relevant risks
6. Expected to improve development quality
7. Widely adopted by other projects or companies
8. Recommended by trusted organizations or institutions
9. Well-documented setup procedures and implementation details
10. Clear prioritization and implementation roadmap
11. Other. (Please specify)

SP9. You can use the security log to monitor activity for your user account and the audit log to monitor activity in your organization. The security and audit log records the type of action, when it was run, and which personal account performed the action. For example, you can use the security log or audit log to track the `org.update_actions_secret` event, which tracks changes to organization secrets. Note: Accessing your security log for user account: (1) In the upper-right corner of any page on GitHub, click your profile photo, then click Settings. (2) In the Archives section of the sidebar, click Security log. Accessing the audit log for organization: (1) In the upper-right corner of GitHub, select your profile photo, then click Your organizations. (2) Next to the organization, click Settings. (3) In the Archive section of the sidebar, click Logs, then click Audit log. Do you regularly audit GitHub Actions using security log or audit log?

1. Yes
2. No
3. Not sure whether we use this practice or not
4. I don't understand what this practice means

SP10. This is a question for those who answered "No" in the previous question. Please select all reasons that apply for not conducting audits of event logs. [Multiple options can be selected.]

1. I was not aware of security logs or audit logs.
2. I do not see the need for auditing.
3. Our team lacks sufficient personnel or time resources.
4. I do not know how to conduct an audit in practice.
5. We do not have the necessary tools or infrastructure in place.
6. Other. (Please specify)

SP11. In your use case, to what extent do you think the platform should autonomously support or take action in auditing GitHub Actions event logs? In this context, “take action” refers to making changes to repository or code settings based on event log analysis. Examples include disabling workflows, revoking tokens, modifying permissions, or enforcing branch protection rules.

1. Level 0: Logging Only (The platform records and stores logs, allowing the user to search and view them manually)
2. Level 1: Decision Support (The platform organizes and visualizes logs to support users in making informed decisions)
3. Level 2: Human Approval (The platform analyzes logs and suggests actions, but requires users to approve those actions before execution)
4. Level 3: Human Veto (The platform automatically analyzes logs and takes action, but users can override or cancel those actions if necessary)
5. Level 4: Execute and Inform (The platform autonomously performs log analysis, decision-making, and actions, and then informs the user of the results)
6. Level 5: Fully Automated (The platform fully autonomously performs log analysis, decision-making, and actions without notifying users)

Questions about Demographics

D1. Please specify your age.

1. 18-19
2. 20-29
3. 30-39
4. 40-49
5. 50-59
6. 60 and over
7. Prefer not to say

D2. What is your gender (Self-identified gender)?

1. Male
2. Female
3. Non-binary
4. Prefer not to say
5. Other. (Please specify)

D3. What is your country of residence?

1. United States
2. India
3. China
4. Brazil
5. United Kingdom
6. Russia
7. Germany
8. Indonesia
9. Japan
10. Canada
11. Prefer not to say
12. Other. (Please specify)

D4. What is the language you are most comfortable with?

1. English
2. Chinese
3. Portuguese
4. Russian
5. German
6. Indonesian
7. Japanese
8. French
9. Prefer not to say
10. Other. (Please specify)

D5. How many years have you been working as a professional software developer ? (Round up if less than one year. If you prefer not to say, enter -1.)

D6. How many years of experience do you have using GitHub Actions? (Round up if less than one year. If you prefer not to say, enter -1.)

D7. What is your primary affiliation when engaging in software development?

1. Employed at a company
2. Freelance/Contract-based
3. Independent developer (personal projects)
4. Prefer not to say

5. Other. (Please specify)

D8. How many employees does your company have? Please select the closest number.

1. 10 or under
2. 11–50
3. 51–100
4. 101–300
5. 301–500
6. 501–1000
7. 1001 or over
8. Not sure
9. Prefer not to say

D9. What is your role in the current company? (Please select the one that most applies to you)

1. Developer, programmer, software engineer
2. Tester / Debugger
3. Project manager
4. Customer support
5. Sales / Marketing
6. Business management
7. Prefer not to say
8. Other. (Please specify)

D10. Please rate your level of confidence in completing the software development tasks given below. [5-point likert scale: "I am not confident at all", "I am slightly confident", "I am somewhat confident", "I am moderately confident," "I am absolutely confident".]

- I can perform a threat risk analysis (e.g., likelihood of vulnerability, impact of exploitation, etc.)
- I can identify potential security threats to the system
- I can identify the common attack techniques used by attackers
- I can identify potential attack vectors in the environment the system interacts with (e.g., hardware, libraries, etc.)
- I can identify common vulnerabilities of a programming language
- I can design software to quarantine an attacker if a vulnerability is exploited
- I can mimic potential threats to the system
- I can evaluate security controls on the system's interfaces/interactions with other software systems

- I can evaluate security controls on the system's interfaces/interactions with hardware systems

Detailed Results of User Study

This section presents the detailed results of the user study. Participant demographics are provided in Table 1. Tables 2 and 3 summarize participants' GitHub repositories and their use of GitHub Actions. Table 4 presents responses on whether each security practice was implemented, and Table 5 lists the reasons for non-implementation. Table 6 summarizes whether participants would be willing to implement the practice after learning about it in this survey. Table 7 presents factors participants consider important when implementing security practices. Table 8 presents reasons for not conducting log audits in GitHub Actions, and Table 9 shows participants' expectations regarding autonomous platform support for log auditing. As part of the qualitative analysis, Tables 10 and 11 list codes developed from open-ended responses on alternative approaches to security practices and other reasons for not implementing the practices.

Table 1 Detailed Participants demographics.

Questions	Options	#
Gender	Male	92
	Female	4
	Non-binary	1
	Prefer not to say	5
Age	18-19	5
	20-29	33
	30-39	31
	40-49	24
	50-59	6
	60 and over	3
Residence	United States	22
	Japan	11
	Germany	8
	France	8
	United Kingdom	6
	Australia	6
	China	4
	Poland	3
	Netherlands	3
	Sweden	3
	Russia	2
	Brazil	2
	Denmark	2
	Canada	2
	Others	14
	Prefer not to say	4
Preferred language	English	52
	French	11
	Japanese	8
	German	7
	Russian	4
	Chinese	4
	Polish	3
	Portuguese	2
	Others	8
	Prefer not to say	2
Primary affiliation when engaging in software development	Employed at a company	47
	Freelance/Contract-based	13
	Independent developer	30
	Others	10
	Prefer not to say	2
Number of employees	10 or under	5
	11–50	12
	51–100	3
	101–300	3
	301–500	2
	501–1000	2
	1001 or over	19
	Prefer not to say	1
Role in the company	Developer, programmer, software engineer	38
	Business management	2
	Project manager	1
	Others	5
	Prefer not to say	1
Development experience (years)		$\mu = 12.3$ $(Md = 8.5)$
Experience using GitHub Actions (years)		$\mu = 4.9$ $(Md = 5.0)$
SSD-SES [?] score		$\mu = 26.3$ $(Md = 27.0)$
# of participants		102

Two participants did not indicate their country of residence, and one participant did not indicate their primary language. The questions on the number of employees and role in the company were asked only to participants who indicated that their primary affiliation when engaging in development is a company.

Table 2 Responses to the Questions about GitHub Repositories that Participants are Involved in Developing.

Questions	Options	#
Contributors	0 (No contributors)	3
	1-4	24
	5-9	18
	10-49	36
	50-99	9
	100-499	11
	500 or over	1
Stars	1-9	3
	10-49	25
	50-99	8
	100-499	30
	500-999	10
	1,000-4,999	18
	5,000-9,999	2
	10,000-49,999	6
# of commits	100-499	14
	500-999	15
	1,000-4,999	49
	5,000-9,999	11
	10,000-49,999	11
	50,000-99,999	1
	100,000 or over	1
Last commit	Within the past 24 hours	41
	Within the past 7 days	31
	Within the past 30 days	16
	Within the past 3 months	8
	Within the past 6 months	4
	Within the past year	1
	More than 1 years ago	1
Repository	User	53
	Organization	49
Role (Organization)	Admin	39
	Organization Member	8
	Outside Collaborator	1
	Contributor	1
Role (User)	Owner	51
	Collaborator	2
Allow GitHub Actions to create/approve PRs	Yes	24
	No	63
	Prefer not to say	1
# of participants		102

The question regarding whether GitHub Actions were allowed to create or approve pull requests was asked only to participants whose role in the repository was Admin or Owner.

Table 3 Response to the Questions about the Use of GitHub Actions.

Questions	Options	#
Automated task by Github Actions (multiple-options)	Build	92
	Testing	83
	Linting	60
	Publishing	48
	Code Scanning	46
	Documentation	44
	Deployment	40
	Formatting	36
	Scheduled Jobs	34
	Dependency Management	32
	Reporting	30
	Container CI	27
	Code Review Support	23
	Security	22
	Project Management	15
	Benchmarking	12
	Localization	6
	Mobile CI	5
	AI Assisted	4
	Monitoring	2
	Other	6
Information sources for configuring GitHub Actions (multople-options)	GitHub official docs	97
	Examples from external repos	64
	Examples from own/org repos	63
	Forums/Q&A websites	59
	Expert web articles	58
	AI tools (e.g., ChatGPT)	36
	Other	3
How serious issue if GitHub Actions becomes unavailable	Very serious	14
	Somewhat serious	52
	Not very serious	32
	Not a problem at all	4
Whether GitHub Actions use sensitive information	Yes	74
	No	27
	Not sure	1
Reusable workflows/Actions are used	Yes	99
	No	3
# of participants		102

Table 4 Responses to the Questions about whether security practices are implemented.

Options	P1	P2	P3	P4-1	P4-2	P5
Yes, I currently implement this practice.	18 (17.6%)	47 (46.1%)	6 (5.9%)	19 (19.4%)	60 (61.2%)	43 (43.4%)
No, I have never implemented this practice.	71 (69.6%)	38 (37.3%)	88 (87.1%)	70 (71.4%)	30 (30.6%)	39 (39.4%)
No, I used to implement this practice but no longer do.	0 (0.0%)	0 (0.0%)	0 (0.0%)	5 (5.1%)	2 (2.0%)	8 (8.1%)
No, but I achieve the same goal using a different approach.	10 (9.8%)	7 (6.9%)	4 (4.0%)	3 (3.1%)	2 (2.0%)	9 (9.1%)
Not sure whether we use this practice or not.	1 (1.0%)	9 (8.8%)	0 (0.0%)	0 (0.0%)	4 (4.1%)	0 (0.0%)
I don't understand what this practice means.	2 (2.0%)	1 (1.0%)	3 (3.0%)	1 (1.0%)	0 (0.0%)	0 (0.0%)
Total	N=102	N=102	N=101	N=98	N=98	N=99

The listed values indicate the number of participants who select each option. The corresponding proportions are shown in parentheses. The number of non-responses for each practice is as follows; P1:0, P2:0, P3:1, P4-1:4, P4-2:4, P5:3.

Table 5 Responses to the Questions about the Reason for not Implementing Security Practices.

Options	P1	P2	P3	P4-1	P4-2	P5
I was not aware of this security practice.	29 (41.4%)	19 (50.0%)	63 (71.6%)	16 (21.3%)	12 (38.7%)	17 (36.2%)
The configuration required to implement this security practice is difficult.	0 (0.0%)	0 (0.0%)	0 (0.0%)	2 (2.7%)	1 (3.2%)	1 (2.1%)
Implementing this practice would increase maintenance or operational costs.	3 (4.3%)	2 (5.3%)	1 (1.1%)	19 (25.3%)	4 (13.0%)	4 (8.5%)
Implementing this practice requires a significant learning effort.	2 (2.9%)	0 (0.0%)	1 (1.1%)	1 (1.3%)	0 (0.0%)	2 (4.3%)
I'm concerned that this practice could negatively impact development.	0 (0.0%)	0 (0.0%)	0 (0.0%)	4 (5.3%)	1 (3.2%)	1 (2.1%)
I don't understand the risks this practice is meant to address or the benefits it provides.	1 (1.4%)	2 (5.3%)	1 (1.1%)	3 (4.0%)	1 (3.2%)	0 (0.0%)
I tried to implement it but gave up due to technical issues (e.g., errors).	0 (0.0%)	0 (0.0%)	0 (0.0%)	1 (1.3%)	0 (0.0%)	1 (2.1%)
I tried to implement it but couldn't get approval or alignment from team members or project leads.	0 (0.0%)	0 (0.0%)	1 (1.1%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
This practice is unnecessary or overly strict for the repository I'm working on.	28 (40.0%)	12 (31.6%)	18 (20.5%)	23 (30.7%)	7 (22.6%)	12 (25.5%)
Other. (Please specify)	7 (10.0%)	3 (7.9%)	3 (3.4%)	6 (8.0%)	5 (16.1%)	9 (19.1%)
Total	N=70	N=38	N=88	N=75	N=31	N=47

The listed values indicate the number of participants who select each option. The corresponding proportions are shown in parentheses. The number of non-responses for each practice is as follows; P1:1, P2:0, P3:0, P4-1:0, P4-2:1, P5:0.

Table 6 Responses to the question of whether previously unaware participants would be willing to implement the practices after learning about them.

Options	P1	P2	P3	P4-1	P4-2	P5
Yes	10 (34.5%)	9 (47.4%)	30 (47.6%)	8 (50.0%)	3 (27.3%)	13 (76.5%)
No	6 (20.7%)	3 (15.8%)	10 (15.9%)	7 (43.8%)	1 (9.1%)	0 (0.0%)
Not Sure	13 (44.8%)	7 (36.8%)	23 (36.5%)	1 (6.3%)	7 (63.6%)	4 (23.5%)
Total	N=29	N=19	N=63	N=16	N=11	N=17

The listed values indicate the number of participants who select each option. The corresponding proportions are shown in parentheses. The number of non-responses for each practice is as follows; P1:0, P2:0, P3:0, P4-1:0, P4-2:1, P5:0.

Table 7 Responses to the Question about Important Factor when implementing security practices (N=102).

Options	# of participants	
Low maintenance and operational overhead	81	(79.4%)
Easy to set up and quick to adopt	47	(46.1%)
Clearly effective in mitigating relevant risks	38	(37.3%)
Does not interfere with the existing development process	33	(32.4%)
Well-documented setup procedures and implementation details	26	(25.5%)
Expected to improve development quality	17	(16.7%)
Low learning curve; easy to understand and use	14	(13.7%)
Widely adopted by other projects or companies	11	(10.8%)
Recommended by trusted organizations or institutions	8	(7.8%)
Clear prioritization and implementation roadmap	1	(1.0%)
Other	2	(2.0%)
Total	102	(100%)

This question allowed multiple selections, with up to three options.

Table 8 Response to the question about reasons not conducting log audits on GitHub Actions (N=80).

Options	# of participants	
I do not see the need for auditing.	36	(45.0%)
I was not aware of security logs or audit logs.	35	(43.8%)
Our team lacks sufficient personnel or time resources.	22	(27.5%)
I do not know how to conduct an audit in practice.	8	(10.0%)
We do not have the necessary tools or infrastructure in place.	6	(7.5%)
Other.	8	(10.0%)
Total	80	(100%)

This question allowed multiple selections. This question was asked only to participants who indicated not performing log audits on GitHub Actions.

Table 9 Response to the Question about Participants Expectation of Platforms' Autonomously Support for Log Audits in GitHub Actions.

Automation level	# of participants	
Level 0: Logging Only	9	(9.2%)
Level 1: Decision Support	14	(14.3%)
Level 2: Human Approval	58	(59.2%)
Level 3: Human Veto	14	(14.3%)
Level 4: Execute and Inform	2	(2.0%)
Level 5: Fully Automated	1	(1.0%)
Total	98	(100%)

Four participants did not answer this question.

Table 10 Codebook for Alternative Approaches to Security Practices.

Code	Example of answer
Alternative tools	[P3] *I manually run zizmor audits on changes.* [P5] *We do the same with Renovate.*
Alternative security feature	[P2] *We restrict permissions for PullRequest-related runs from untrusted sources.* [P4-2] *Tags are no longer used at all for actions, only commit hashes.*
Enforce Review	[P1] *We always require code owners to inspect Pull Requests, [...]* [P1] *I enforce 1-2 reviews from GitHub org members.*
Other	

The ID in brackets indicates which security practice the answer example corresponds to.

Table 11 Codebook for reasons for not implementing security practices.

Code	Example of answer
Not applicable	[P1] *As single owner, this step is unnecessary. [...]* [P2] *I don't use external variables nor any substantial non-linear code in my actions.* [P4-1] *I use only GitHub actions avoiding 3rd-party actions.* [P5] *Dependabot doesn't support C++*
Negative impact on development	[P1] *Some of my collaborators are not as GitHub savvy and might get upset if something like this inhibited their ability to contribute.* [P5] *we don't want to increase deps without deeper discussions with community or the team so this is sure a manual process.*
Negative impact on security	[P4-1] *This could make us more vulnerable by not automatically using the latest version of an action that might have important security fixes.* [P5] *Rubber stamp updating the dependencies violates the point of pinning to a commit.*
Lack of resources	[P4-1] *I'm willing to do this, but I have no time at the moment.*
Other	

The ID in brackets indicates which security practice the answer example corresponds to.