

Administration of GitHub Advanced Security (GHAS)

Notes, SOP, and Example Markdown Templates

Learning Objectives

After completing this guide, you should be able to:

- Explain how GHAS features are enabled across enterprise, org, and repo scopes.
- Set org-level defaults and repo overrides with least-privilege RBAC.
- Use APIs and the `gh` CLI for automation and reporting.
- Standardize community health files and workflow templates via `.github`.

1 `.github` Org Repository

Suggested Structure

```
1 .github/  
2   CODE_OF_CONDUCT.md  
3   CONTRIBUTING.md  
4   SECURITY.md  
5   FUNDING.yml  
6   ISSUE_TEMPLATE/  
7     bug_report.md  
8     feature_request.md  
9   workflow-templates/  
10     codeql.yml  
11     codeql.properties.json  (template metadata)
```

2 Permissions, APIs, and CodeQL (Quick Reference)

List Code Scanning Alerts (repo scope)

```
1 gh api repos/OWNER/REPO/code-scanning/alerts --paginate --jq '.[].rule.id'
```

List Dependabot Alerts (org scope)

```
1 curl -s -H "Authorization: Bearer $GITHUB_TOKEN" \  
2   -H "Accept: application/vnd.github+json" \  
3   "https://api.github.com/orgs/ORG/dependabot/alerts?per_page=100"
```

List Secret Scanning Alerts (repo scope)

```
1 gh api repos/OWNER/REPO/secret-scanning/alerts --paginate
```

Baseline CodeQL Workflow (Template)

```
1 name: CodeQL  
2 on:  
3   push: { branches: ["main"] }  
4   pull_request: { branches: ["main"] }  
5   schedule: [ { cron: "23 3 * * 1" } ]  
6  
7 jobs:  
8   analyze:  
9     runs-on: ubuntu-latest  
10    permissions:  
11      security-events: write  
12      contents: read  
13      actions: read  
14    steps:  
15      - uses: actions/checkout@v4  
16      - uses: github/codeql-action/init@v3  
17        with:  
18          languages: auto  
19          # queries: security-extended,security-and-quality  
20      - uses: github/codeql-action/autobuild@v3  
21      - uses: github/codeql-action/analyze@v3
```

3 Markdown File Examples

This section provides copy-pasteable starting points for each Markdown file in the structure above. Replace placeholders like `ORG`, `CONTACT_EMAIL`, and `SECURITY_TEAM_URL`.

3.1 `CODE_OF_CONDUCT.md`

```
1  # Code of Conduct
2
3  ## Our Pledge
4  We as members, contributors, and leaders pledge to make participation in our
5  community a harassment-free experience for everyone.
6
7  ## Our Standards
8  Examples of behavior that contributes to a positive environment include:
9  - Demonstrating empathy and kindness
10 - Being respectful of differing viewpoints and experiences
11 - Accepting constructive feedback gracefully
12
13 Examples of unacceptable behavior include:
14 - Harassment, trolling, or derogatory comments
15 - Publishing others' private information without permission
16 - Other conduct which could reasonably be considered inappropriate
17
18 ## Enforcement Responsibilities
19 Project maintainers are responsible for clarifying and enforcing our standards.
20 They will take appropriate and fair corrective action in response to any behavior
21 they deem inappropriate, threatening, offensive, or harmful.
22
23 ## Scope
24 This Code applies within project spaces and in public spaces when an individual
25 is representing the project or its community.
26
27 ## Enforcement
28 Instances of abusive, harassing, or otherwise unacceptable behavior may be
29 reported to the maintainers at CONTACT_EMAIL. All complaints will be reviewed
30 and investigated promptly and fairly.
31
32 ## Attribution
33 This Code of Conduct is adapted from common open-source community guidelines.
```

3.2 CONTRIBUTING.md

```
1 # Contributing to ORG
2
3 Thanks for taking the time to contribute! The following is a set of guidelines
4 for contributing to projects under the ORG organization.
5
6 ## Ground Rules
7 - Follow our [Code of Conduct](../CODE_OF_CONDUCT.md).
8 - Discuss significant changes via issues before opening a PR.
9 - Keep PRs focused and small when possible.
10
11 ## Development Workflow
12 1. Fork the repo and create a feature branch:
13    - `git checkout -b feat/short-description`
14 2. Run tests and linters locally.
15 3. Commit using conventional commits where possible.
16 4. Open a PR against `main` and fill out the PR template.
17
18 ## Commit Message Guidelines (Conventional Commits)
19 - `feat: add new user profile endpoint`
20 - `fix: correct null pointer in auth middleware`
21 - `docs: update README for setup`
22 - `ci: bump CodeQL version to v3`
23
24 ## Pull Request Checklist
25 - [ ] Tests added or updated
26 - [ ] Docs updated (README/SECURITY/CHANGELOG as applicable)
27 - [ ] CI is green
28 - [ ] No secrets or credentials included
29
30 ## Issue Labels
31 - `bug`, `feature`, `documentation`, `security`, `good first issue`
32
33 ## Licensing
34 By contributing, you agree that your contributions will be licensed under the
35 project's license.
```

3.3 SECURITY.md

```
1 # Security Policy
2
3 ## Supported Versions
4 We generally support security fixes for the latest release and the most recent
5 minor version line. Older releases may receive fixes at our discretion.
6
7 ## Reporting a Vulnerability
8 Please email CONTACT_EMAIL or use our private intake form:
9 SECURITY_TEAM_URL
10
11 **Do not** open public issues for suspected vulnerabilities.
12
13 Provide as much detail as possible:
14 - Affected version(s) and component(s)
15 - Reproduction steps and proof of concept
16 - Impact assessment (confidentiality, integrity, availability)
17 - Any suggested mitigations
18
19 We will acknowledge receipt within 2 business days and provide regular updates
20 until resolution.
21
22 ## Disclosure Policy
23 We follow coordinated disclosure. After a fix is available and users have had
24 a reasonable window to update, we may publish details in release notes.
25
26 ## Security Hardening Guidance
27 - Enable branch protection and required status checks
28 - Use GHAS features (secret scanning, Dependabot, CodeQL)
29 - Rotate credentials regularly and prefer OIDC for CI
```

3.4 ISSUE_TEMPLATE/bug_report.md

```
1 ---
2 name: "Bug report"
3 about: "Create a report to help us improve"
4 title: "[Bug]: short summary"
5 labels: ["bug"]
6 assignees: []
7 ---
8
9 ## Describe the bug
10 A clear and concise description of the problem.
11
12 ## To Reproduce
13 Steps to reproduce the behavior:
14 1. Go to '...'
15 2. Run '...'
16 3. Observe '...'
17
18 ## Expected behavior
19 What you expected to happen.
20
21 ## Screenshots or logs
22 If applicable, add screenshots or logs.
23
24 ## Environment
25 - OS: e.g., Ubuntu 22.04
26 - Runtime/SDK: e.g., Node.js 20.x
27 - Version/Commit: e.g., v1.2.3 / abc123
28
29 ## Additional context
30 Add any other context about the problem here.
```

3.5 ISSUE_TEMPLATE/feature_request.md

```
1 ---
2 name: "Feature request"
3 about: "Suggest an idea for this project"
4 title: "[Feature]: short summary"
5 labels: ["feature"]
6 assignees: []
7 ---
8
9 ## Motivation / Problem Statement
10 Is your request related to a problem? Please describe it clearly.
11
12 ## Proposed Solution
13 Describe the solution you'd like, including API or UX changes if relevant.
14
15 ## Alternatives
16 Describe any alternative solutions or features you've considered.
17
18 ## Acceptance Criteria
19 - [ ] Clear, testable criteria 1
20 - [ ] Clear, testable criteria 2
21 - [ ] Docs updated
22
23 ## Additional context
24 Add any other context or mockups here.
```

4 Policy and Inheritance Cheatsheet

Level	What to Prefer Centralizing	Typical Overrides
Enterprise	Global minimums/guardrails, audit, org defaults	Org-specific constraints
Organization	Default GHAS enablement, templates, required checks	Repo languages, schedules, build steps
Repository	Local exceptions, custom queries/build	N/A (justify exceptions)

5 Operational Playbook

1. Establish governance: .github repo, community health files, templates.
2. Enable GHAS org-wide: secret scanning, Dependabot, CodeQL.
3. RBAC: Security teams with least-privilege permissions.
4. Triage and SLAs: severity, dismissal reasons, escalation paths.
5. Automate: APIs and gh for alert export, dashboards, tickets.
6. Measure: MTTR, reopened rates, coverage, query packs.