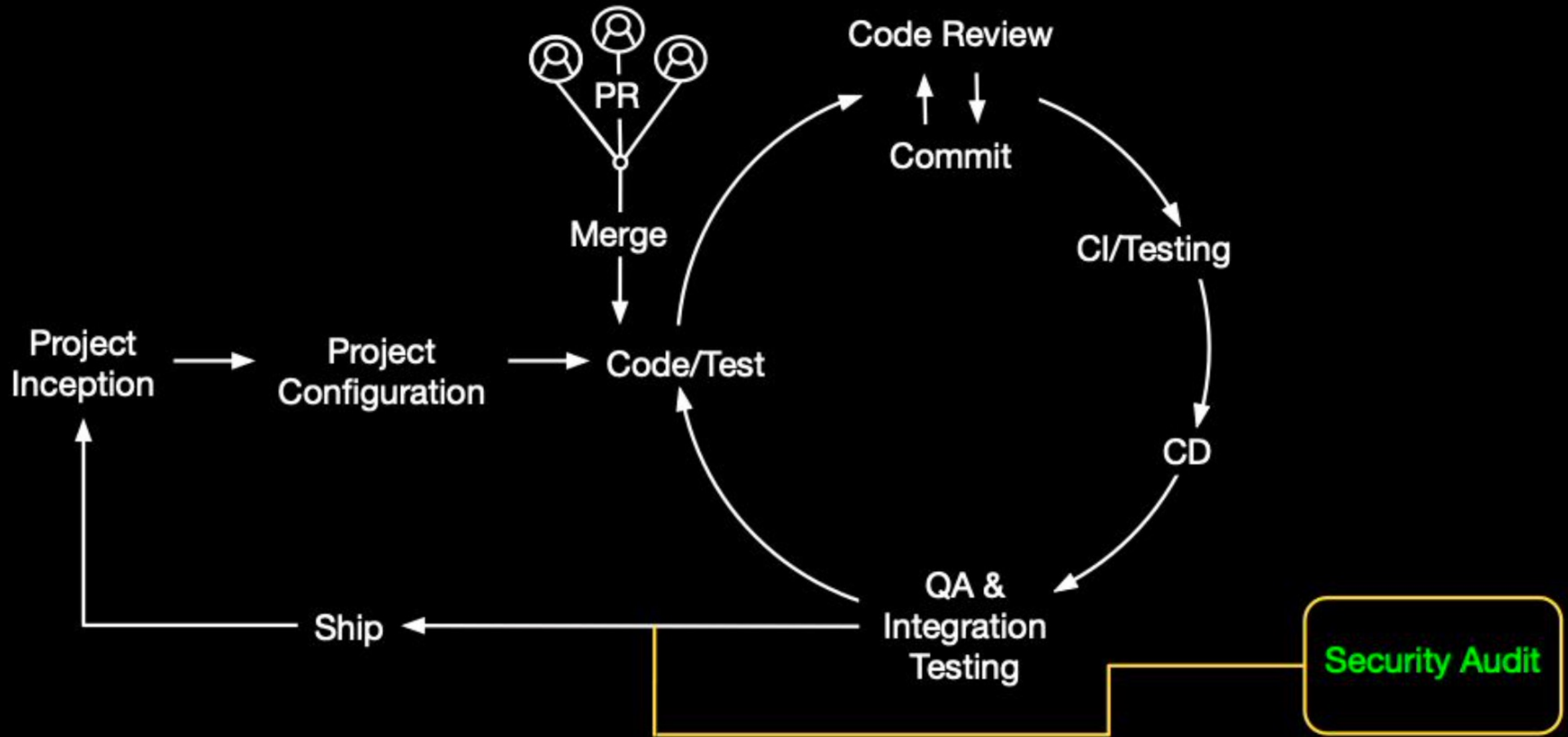


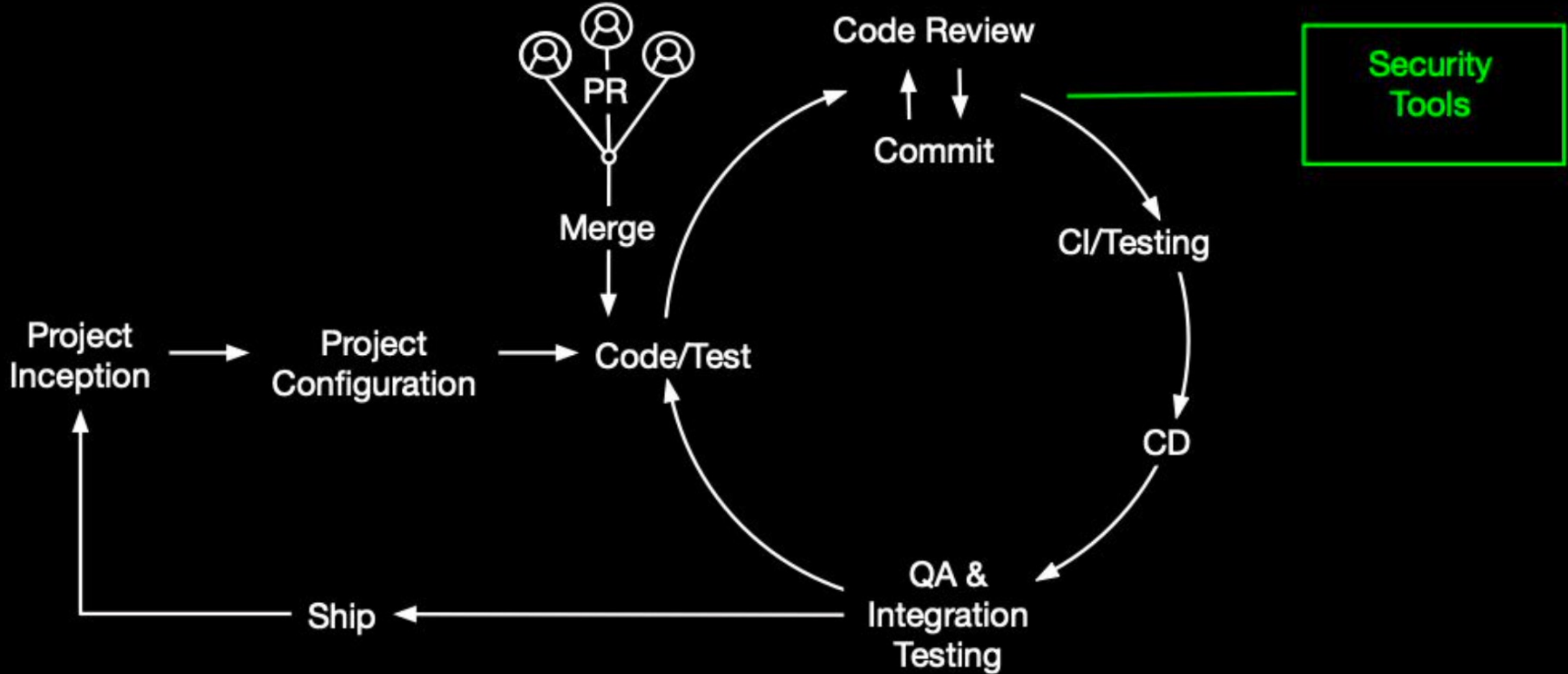
## Customer Scenario

"Our security team is asking for help ensuring proper reviews are being done to code being added into our repositories. We have hundreds of repositories in our organization. What is the best way we can achieve at scale? We are new to some of the out-of-the-box settings and the GitHub API. Can you please help us create a solution that will accomplish this for our security team?"

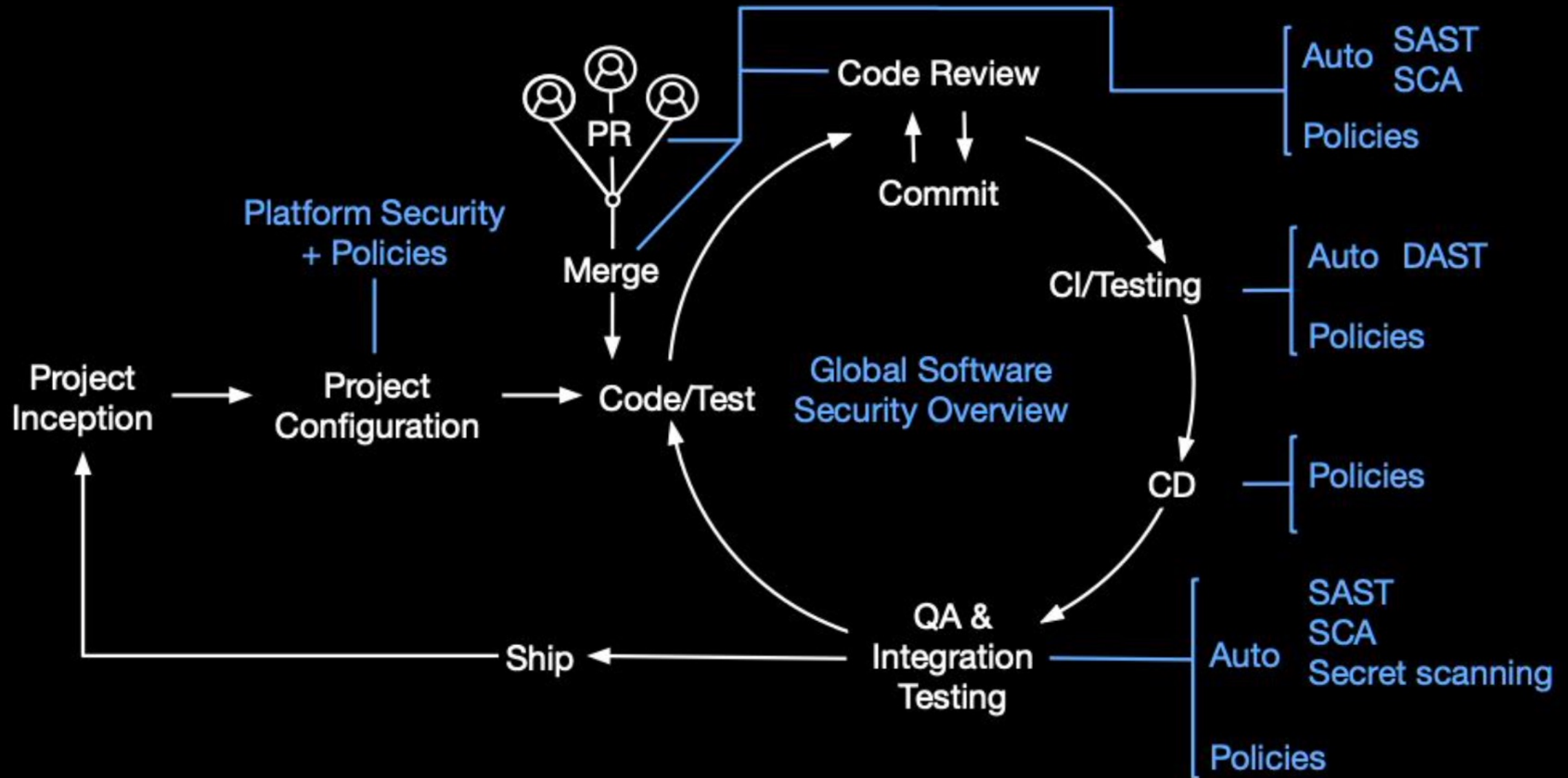
# Basic application security scenario



# Improved application security scenario



# Application security - Targeted State



# **GitHub delivers complete application security**

- 1. Developer-first**
- 2. Native**
- 3. Automated**

# Secure your source code

**Fail fast by finding vulnerabilities as code is developed**

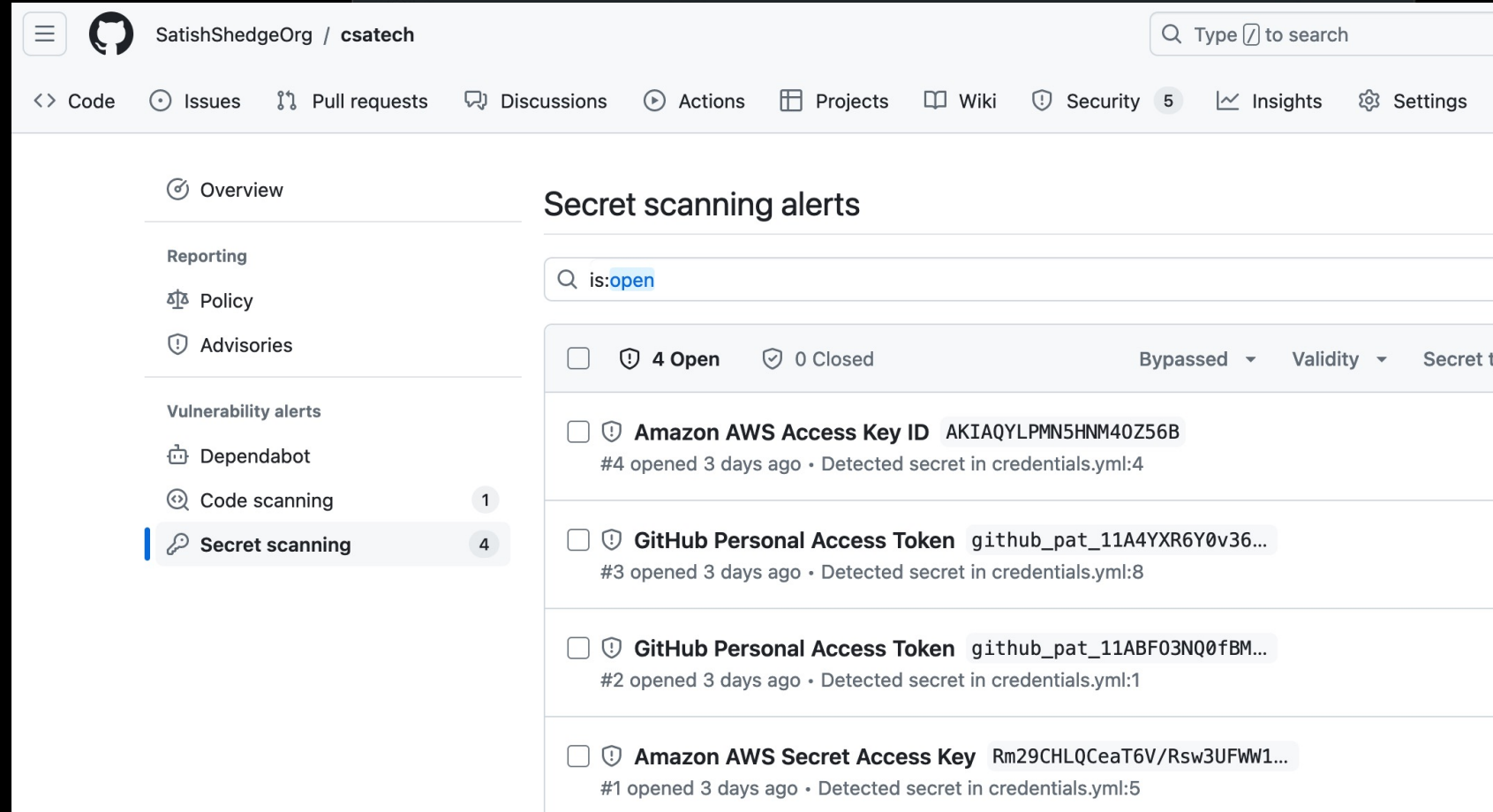
**Find hard-coded secrets in code base**

**Set coding standards across your entire organisation**

**Know your environment**

**Manage your dependencies**

**Fix and publish vulnerability information**



The screenshot displays the GitHub Security Center interface for the repository 'SatishShedgeOrg / csatech'. The left sidebar contains navigation links: Overview, Reporting, Policy, Advisories, Vulnerability alerts, Dependabot, Code scanning (1), and Secret scanning (4). The main content area is titled 'Secret scanning alerts' and shows a search bar with 'is:open'. Below the search bar, there are four alerts, each with a checkbox, a warning icon, a title, a secret value, and a description. The alerts are: 1. Amazon AWS Secret Access Key (Rm29CHLQCeAT6V/Rsw3UFWW1...), 2. GitHub Personal Access Token (github\_pat\_11ABF03NQ0fBM...), 3. GitHub Personal Access Token (github\_pat\_11A4YXR6Y0v36...), and 4. Amazon AWS Access Key ID (AKIAQYLPMN5HNM40Z56B). All alerts were detected in 'credentials.yml' and opened 3 days ago.

Alert ID	Secret Type	Secret Value	Location	Status
#1	Amazon AWS Secret Access Key	Rm29CHLQCeAT6V/Rsw3UFWW1...	credentials.yml	Open
#2	GitHub Personal Access Token	github_pat_11ABF03NQ0fBM...	credentials.yml	Open
#3	GitHub Personal Access Token	github_pat_11A4YXR6Y0v36...	credentials.yml	Open
#4	Amazon AWS Access Key ID	AKIAQYLPMN5HNM40Z56B	credentials.yml	Open

# High-Level Solution Summary

Customer should leverage GitHub advance security features, along with some of other best practices of shift security left with fail fast by finding vulnerabilities, hard coded secrets and by setting code standards

- Code Security configuration would help with Secret scanning and also blocking commits by Push Protection
- Code scanning with CodeQL default configuration automatically detects vulnerabilities and coding errors
- Dependency graph would give details about vulnerabilities with dependencies and also auto remediation by security updates.
- Private vulnerability reporting: Allow your community to privately report potential security vulnerabilities in public repositories
- GitHub API and webhook to leverage alerts integration
- Repository Rulesets for enforcing Branch protection
- Organization Team and events for notifying team or individual about code review

# GitHub Advanced Security features has below solutions for challenges faced by customer

## Repository Security

- **Private repositories:** Host code that you don't want to share with the world in private GitHub repositories only accessible to you and people you share them with.
- **Repository rules** Elevate your organization's security with source code protections that scale. Rule insights make it easy to review how and why code changed in your repositories.

**Code scanning:** Secure code as you write it. Automatically review every change to your codebase and identify vulnerabilities before they reach production.

**Secret scanning:** Automatically detect and deactivate secrets committed to your repos. Receives alerts when secrets or keys are checked in and defines custom patterns.

**Dependency review:** Shows the full effect of changes to dependencies and sees details of any vulnerable versions before you merge a pull request.

**GitHub Advisory Database** Browse or search for the vulnerabilities that GitHub knows about. The database contains all curated CVEs and security advisories on the GitHub dependency graph.



# Security built into the Developer Lifecycle

## Benefits of Organization level Code Security configuration

- Dependency graph: Display license information and vulnerability severity for your dependencies. Always enabled for public repositories.
- Dependabot: Receive alerts for vulnerabilities that affect your dependencies
- Security updates: Allow Dependabot to open pull requests automatically to resolve alerts
- Code scanning: Receive alerts for automatically detected vulnerabilities and coding errors using CodeQL default configuration
- Secret scanning: Receive alerts for detected secrets, keys, or other tokens
- Push protection: Block commits that contain supported secrets
- Private vulnerability reporting: Allow your community to privately report potential security vulnerabilities in public repositories

# GitHub API and Security Alerts

## Code Scanning Alerts:

- CodeQL Analysis Alerts: Generated by CodeQL, GitHub's semantic code analysis engine, these alerts identify potential security vulnerabilities in the codebase. They cover a wide range of issues, including but not limited to SQL injection, cross-site scripting, and other code vulnerabilities.

## Secret Scanning Alerts:

- These alerts are triggered when potentially sensitive information, such as API keys or credentials, is identified within the repository's source code.

## Dependency Alerts:

- Dependabot automatically detects outdated dependencies in a project and creates pull requests to update them to the latest, secure versions.

## Security Overview Alerts:

- The Security Overview provides a comprehensive dashboard summarizing the security status of the repository.

## Third Party Alerts:

- Integrate third-party code analysis tools with GitHub code scanning by uploading data as SARIF files.

[GitHub Security Overview](#)

[Dependabot](#)

[Code scanning](#)

[Secret scanning](#)

[CodeQL](#)

[Integrating with code scanning](#)

[GitHub security features](#)

[About GitHub Advanced Security](#)

[About code scanning](#)

[About secret scanning](#)

[About dependency review](#)

[About security overview](#)

[Enabling or disabling a feature for all existing repositories](#)

[Enabling GitHub Advanced Security for your enterprise](#)

[Enforcing a policy for the use of GitHub Advanced Security in your enterprise](#)

[Adding a security policy to your repository](#)

[REST API](#)

[GraphQL API](#)

[Automatic token authentication](#)

