Pull What Where?

Contributing to Open Source, Securely, on GitHub
GitHub for over three years
✴ Casual Open Source enthusiast
✴❤ The outdoors and landscape photos

Jamie Jones

GitHub Principal Architect & Regulated Industries Solutions Lead
✴ GitHub for over three years
✴ Casual Open Source enthusiast
✴❤ The outdoors and landscape photos

@jbjonesjr
Open Source Readiness Program
- Regulatory Compliance WG Chair

FinServ Developer Experience Program Chair
- Developer Experience WG Chair
Goals

- Overview of the value of open source
- Understanding the different workflows used publishing code outside your organization
- Understanding GitHub and integration points key for process security
- Better practices for open source security
- Things to think about when staying legal and compliant with open source code
- Critical tools to integrate into your process
“Ten or 15 years ago, the engineers were the ones who didn’t speak to anyone and maybe seemed like they hadn’t showered that day. The traders were the ones that looked like they stepped out of a Brooks Brothers catalog,” said. “That line has basically disappeared.”

– Oliver Cooke, a financial-industry recruiter at Selby Jennings

“Everyone who comes to sales and trading needs to know how to code.”

– Adam Korn, Goldman Sachs
OSS usage continues to grow

What percent of the typical app you work on is comprised of open source components?

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Percentage of Apps</th>
<th>Total (in millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-49%</td>
<td>64.4%</td>
<td>13,960,231</td>
</tr>
<tr>
<td>50-74%</td>
<td>27.8%</td>
<td>6,122,356</td>
</tr>
<tr>
<td>75-100%</td>
<td>8.8%</td>
<td>1,935,860</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>22,018,447</td>
</tr>
</tbody>
</table>

What percentage of Apps include ANY open source?

96%
Top open source organizations

- Microsoft: 7700
- Google: 5500
- Red Hat: 3300
- UC Berkeley: 2700
- Intel: 2200
- Univ. of Washington: 1800
- Facebook: 1700
2017 - Open Source in Financial Services

23,000 Contributors to 43,000 projects totaling 350,000 contributions

26% Contributor growth YoY

40% Contribution growth YoY

💰💰💰
Nearly 50% of the current S&P 500 will be replaced by 2026

Innosight
Corporate Longevity: Turbulence Ahead for Large Organizations
Open Source Workflows
Code in the open!!!!

GitHub.com
Do it on your own time!

- Process enforcement
- Leadership Approvals
- Security
  - Pii, Dependency Checking,
  - Dirty Words, Token Scanning,
  - Static and Dynamic inspection,
  - ALMSS, Vulnerability tracking
- Compliance & Audit
Okay, we’ll integrate… with a process

- Filling out forms
- Getting approvals
- Waiting days (or weeks)
- Burning media… lots of media
- What will legal say?
- Whose even investigating these things, anyway?

GitHub.com
What if we used a proxy?

- Process enforcement
- Leadership Approvals
- Security
  - Pii, Dependency Checking,
  - Dirty Words, Token Scanning,
  - Static and Dynamic inspection,
  - ALMSS, Vulnerability tracking
- Compliance & Audit

GitHub.com
A better process? maybe?

- Filling out forms
- Getting approvals
- Waiting days (or weeks)
- Burning media… lots of media
- What will legal say?
- Whose even investigating these things, anyway?
Where we are going

- Process enforcement
- Leadership Approvals
- Security
  - Pii, Dependency Checking,
  - Dirty Words, Token Scanning,
  - Static and Dynamic inspection,
  - ALMSS, Vulnerability tracking
- Compliance & Audit

GitHub.com
Code in the open!!!!

GitHub.com
GitHub Security Best Practices
Vulnerability Scanning

Data services

Use the data from your repository to power these enhanced features. If you’d like to enable the dependency graph, vulnerability alerts, and services like it, we’ll need additional permissions.

- **Allow GitHub to perform read-only analysis of this repository**
  By checking the “Allow GitHub to perform read-only analysis of this repository” checkbox, you’re agreeing to GitHub’s Terms of Service and granting us permission to perform read-only analysis of this private repository. Learn more about how we use your data.

- **Dependency graph**
  Access My_Project’s dependencies, sub-dependencies, versions, and related repositories on GitHub.

- **Vulnerability alerts**
  Receive alerts for known security vulnerabilities found in dependencies.
Alert Notification

Alerts

⚠️ 1 Open  ✔️ 0 Closed

⚠️ hoek
opened 9 hours ago by GitHub + package-lock.json

GitHub tracks known security vulnerabilities in some dependency manifest files. Learn more about alerts.
Protected Branches

Branch protection rule

Apply rule to

Rule settings

Protect matching branches
Disables force-pushes to all matching branches and prevents them from being deleted.

- Require pull request reviews before merging
  When enabled, all commits must be made to a non-protected branch and submitted via a pull request with
  the required number of approving reviews and no changes requested before it can be merged into
  a branch that matches this rule.

- Require status checks to pass before merging
  Choose which status checks must pass before branches can be merged into a branch that matches this
  rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to
  a branch that matches this rule after status checks have passed.

- Require signed commits
  Commits pushed to matching branches must have verified signatures.

- Include administrators
  Enforce all configured restrictions for administrators.

- Restrict who can push to matching branches
  Specify people or teams allowed to push to matching branches. Required status checks will still prevent
  these people from merging if the checks fail.
Enforce compliance policies with **branch statuses**, **branch protection**, **required reviews** and inline conflict resolution.
Restrict access

Enforce all configured restrictions for administrators.

**Restrict who can push to matching branches**
Specify people or teams allowed to push to matching branches. Required status checks will still prevent these people from merging if the checks fail.

- jbjon
- jbjonjr Jamie Jones

**People and teams with push access**

- **Organization and repository administrators**
  These members can always push.

- hubot
  Hubot
Code Owners

> cat CODEOWNERS

# Lines starting with '#' are comments.
# Each line is a file pattern followed by one or more owners.

# These owners will be the default owners for everything in the repo.
*       @defunkt

# Order is important. The last matching pattern has the most precedence.
# So if a pull request only touches javascript files, only these owners
# will be requested to review.
*.js    @octocat @github/js

# You can also use email addresses if you prefer.
docs/*  docs@example.com
Cleanup after yourself

Danger Zone

Make this repository public
Make this repository visible to anyone.

Transfer ownership
Transfer this repository to another user or to an organization where you have the ability to create repositories.

Archive this repository
Mark this repository as archived and read-only.

Delete this repository
Once you delete a repository, there is no going back. Please be certain.
Bonus: signed commits for additional assurance

Rule settings

Protect matching branches
Disables force-pushes to all matching branches and prevents them from being deleted.

Require pull request reviews before merging
When enabled, all commits must be made to a non-protected branch and submitted via a pull request with the required number of approving reviews and no changes requested before it can be merged into a branch that matches this rule.

Require status checks to pass before merging
Choose which status checks must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

Require signed commits
Commits pushed to matching branches must have verified signatures.

Include administrators
Enforce all configured restrictions for administrators.

Restrict who can push to matching branches
Specify people or teams allowed to push to matching branches. Required status checks will still prevent these people from merging if the checks fail.
Open Source Security & Compliance Best Practices
Dependency Management

• Dependency Management tools like
  • greenkeeper.io, Dependabot, Depfu, Renovate

• …Or check nightly via your own custom scripts
  • Download, run dependency update per language spec, use diff to create patch
Container and image scanning

- You can’t swing a dead cat without hearing about scanning solutions:
  - Kritis, Clair, AquaSec, Sonatype, BlackDuck, OpenSCAP

- Important to remember that container scanning is not THE security process, but a piece of it
Vulnerability data

• While GitHub includes data from the NVD system and our own platform research, there are numerous other providers of data

• Snyk
• Whitesource
• BlackDuck
• Sonatype

• All leading to… your OWN threat intelligence
Build often

• Don’t just run builds after commits or merges, but on routine basis as well!

• You never know when your bugs will appear
Static, Dynamic, and Browser Testing! Oh My!

- Static Code Analysis Tools like SonarQube, Codacy, WhiteSource, OWASP, Fortify, Veracode, Coverity
- Dynamic Code Analysis Tools like Veracode, Selenium, OWASP ZAP
Document and Security and Compliance policy

• We talk about what licenses we allow, but do we plan for the vulnerabilities we allow?

• What about the flavor of contributor we encourage?

• Are their activity requirements on the dependencies? Do they have to be active at all?
Open Source tools and Integrations
<table>
<thead>
<tr>
<th>Category</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vulnerability tools</td>
<td>Whitesource, Veracode, Sonatype, IonChannel, GitHub</td>
</tr>
<tr>
<td>Static Code Analysis</td>
<td>SonarQube, Selenium, Fortify, Codacy, OWASP, Coverity</td>
</tr>
<tr>
<td>Dynamic Code Analysis</td>
<td>Veracode, Selenium, OWASP ZAP</td>
</tr>
<tr>
<td>Dependency Management</td>
<td>Dependabot, greenkeeper.io, Renovate, Depfu</td>
</tr>
<tr>
<td>Container Scanning</td>
<td>Kritis, Clair, AquaSec, Sonatype, BlackDuck, OpenSCAP</td>
</tr>
<tr>
<td>Vulnerability Resources</td>
<td>Whitesource, Sonatype, Snyk, BlackDuck</td>
</tr>
</tbody>
</table>
Open Developer Platform

Hosted Platforms

- SYMPHONY (ReST API)
- SYMPHONY (Extension API)
- SYMPHONY (Integration webhooks)
- FINTECH OPEN APIs
- CLOUD OPEN APIs
- FINTECH OPEN DATA

Future partnerships and contributions

High Productivity Turnkey Developer Experience

Development Infrastructure

- OPEN SOURCE DEVELOPER
  - CODE HOSTING: Github
  - CI: Travis CI
  - SECURITY & LEGAL CHECKS: Whitesource
  - CD: OpenShift
  - RELEASE: Maven central, NPM, NuGet

Biz & Legal Peace Of Mind - We Do The Hard Part!

Collaboration Services

- WIKI: Atlassian Confluence
- MAILING LISTS: Google Groups
- WEB CONFERENCING: WebEx
- METRICS & REPORTING: Bitergia
Recap

• Understanding the different workflows used publishing code outside your organization
• Understanding GitHub and integration points key for process security
• Better practices for open source security
• Things to think about when staying legal and compliant with open source code
• Critical tools to integrate into your process
Code Thoughts™ for Open Source participation
How are you managing a community?
Where do they exist today? How are you going to meet them there?
What happens when someone acts inappropriately? What’s your PR strategy?
Everything (EVERYTHING) in the Core Infrastructure Initiative Best Practice program
Code Thoughts™ v2 - With more security

• Who secures your projects?
  • Your internal security team? `The community`? Who is responsible for those critical fixes? But what if YOU’RE not there?
• Do you even audit?
• Do you have an incident response plan for severe vulnerabilities?
  • Where do you report? Do you?
• [https://github.com/envoyproxy/envoy/blob/master/SECURITY_RELEASE_PROCESS.md](https://github.com/envoyproxy/envoy/blob/master/SECURITY_RELEASE_PROCESS.md)
Questions and Answers?