GitHub

How GitHub secures open source software

Learn how GitHub works to protect you as you use, contribute to, and build on open source.

GitHub's role in securing open source software

Open source software is everywhere, powering the languages, frameworks, and applications your team uses every day.

A study conducted by the Synopsys Center for Open Source Research and Innovation found that enterprise software is now comprised of more than 90 percent open source code—and businesses are taking notice. The State of Enterprise Open Source study by Red Hat confirmed that "95 percent of respondents say open source is strategically important" for organizations. Making code widely available has changed how software is built, with more reuse of code and complex dependencies—but not without introducing security and compliance concerns. Open source projects, like all software, can have vulnerabilities. They can even be the target of malicious actors who may try to use open source code to introduce vulnerabilities downstream, attacking the software supply chain. These threats expose your organization to additional risk.

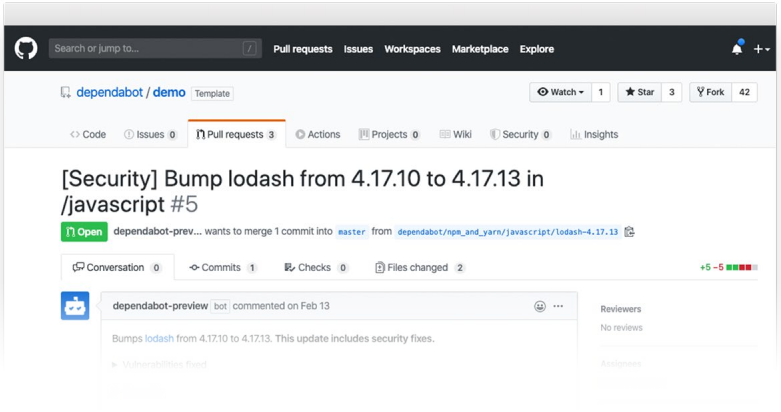
At GitHub, we see security as an issue we need to address as a community: one that affects all software, regardless of how much proprietary code it contains. Similarly, a safe and healthy open source community isn't just good for open source software. It also benefits the millions of businesses that depend on it.

That's why we've built tools and processes that allow organizations and open source maintainers to code securely throughout the entire software development lifecycle. Taking security and shifting it to the left allows organizations and projects to prevent errors and failures before a security incident happens.

GitHub works hard to secure our community and the open source software you use, build on, and contribute to. Through features, services, and security initiatives, we provide the millions of open source projects on GitHub—and the businesses that rely on them—with best practices to learn and leverage across their workflows.

Making open source more secure

GitHub Advisory Database, vulnerable dependency alerts, and Dependabot



One of the key elements of identifying security issues is working with a rich database of vulnerabilities. GitHub's dependency vulnerability detection tools use a combination of data directly from GitHub Security Advisories and the National Vulnerability Database (NVD) to create a complete picture of vulnerabilities in open source. This combined dataset lives in the GitHub Advisory Database and powers Dependabot alerts and security updates. The Advisory Database is also under a Creative Commons Attribution 4.0 license, meaning it's freely available for anyone to use as long as they attribute GitHub as the data source.

To generate automated fixes, we start with CVE alerts, which describe vulnerable and remediated versions, then identify susceptible repositories using their respective language dependency management definitions. This allows us to parse a repository's manifest and alert their administrators to vulnerable dependencies and, specifically, to the versions they need to update to in order to remediate these issues. Since the launch of security alerts in 2017, we've sent alerts on more than 89 million vulnerabilities found in open source repositories.

89,125,270

Security vulnerability alerts sent (as of March 18, 2020)

Although similar capabilities are available in third-party tools, our research shows that many open source repositories don't take full advantage of them. Being notified about vulnerabilities via alerts is only part of solving the problem. Our data also shows that of those 89 million vulnerability alerts, a surprising 70 percent remained unfixed a month post notification. Although they're now aware a vulnerability exists, many developers and administrators aren't sure how to resolve it—leaving their applications open to security issues and attacks. To help, GitHub suggests automatic fixes. Once a vulnerability is found, we use Dependabot to automatically create a pull request for known vulnerabilities that allow you to quickly merge and deploy remediating changes to your codebase. That means each security alert now includes a severity level, a link to the affected file in your project, and a link to a pull request with the automated security fix.

GitHub also makes it easier for open source project maintainers to address and share newly-found security vulnerabilities. Rather than relying on mailing lists, open source groups, release notes, or changelogs to communicate with their users, projects can now communicate directly on GitHub. GitHub is a CVE Numbering Authority (CNA) and is authorized to assign CVE identification numbers. Thanks to this capability, maintainers can request a CVE number for an issue they're dealing with, and publish information directly on GitHub so that developers within the GitHub community see the advisory first. This allows us to generate security

vulnerability alerts for the entire community and share them first on GitHub.

But maintainers don't always mark the security fixes they find—so finding these commits among the vast number of GitHub processes every day requires some extra help. Our machine learning model sifts through all commits on dependency files and extracts the ones that might be related to a security release. The model uses diffs and commit messages to learn how the required version range changed and understand the intent of the change. Then it aggregates over time to determine if a dependency has released a new version with a security fix that should trigger an alert.

What's next

Today, GitHub supports security alerts and automated security updates for NuGet, Python, Ruby, npm, Yarn, Composer and Maven ecosystems—but any open source maintainer can publish an advisory on GitHub as it applies to their project. We'll continue adding internally and externally sourced updates to our vulnerability data, helping projects better understand their risks. And we plan to continue adding support for more ecosystems.

Responsible disclosure and access

Despite code scanning and protection from malicious actors, vulnerabilities will inevitably be found. And when they are, GitHub makes vulnerability disclosure and management as simple as possible.

To start, we released our Security Advisory API to provide security advisories as a public service. A building block toward a powerful security platform, it provides a way to access the security feeds we aggregate and validate, plus the dependency upgrades we monitor across millions of projects. Now vulnerability data is easily available and ready to be integrated into the tools and workflows you already use.

The Security Advisory API also provides additional capabilities and complements the NVD feeds with concerns like malware and other vulnerabilities that GitHub Security Lab has found and shared. As a public service, the API provides a foundation for GitHub, researchers, and integrators to collectively create more secure software for all of us.

In order to improve the security of the open source supply chain, maintainers can fix and disclose the existence of vulnerabilities by publishing advisories in their GitHub repositories. Advisories created through this feature are then curated through our internal advisory database service and broadcast through dependency alerts, our external database, and the API. Users can also request and be assigned CVE identifiers.

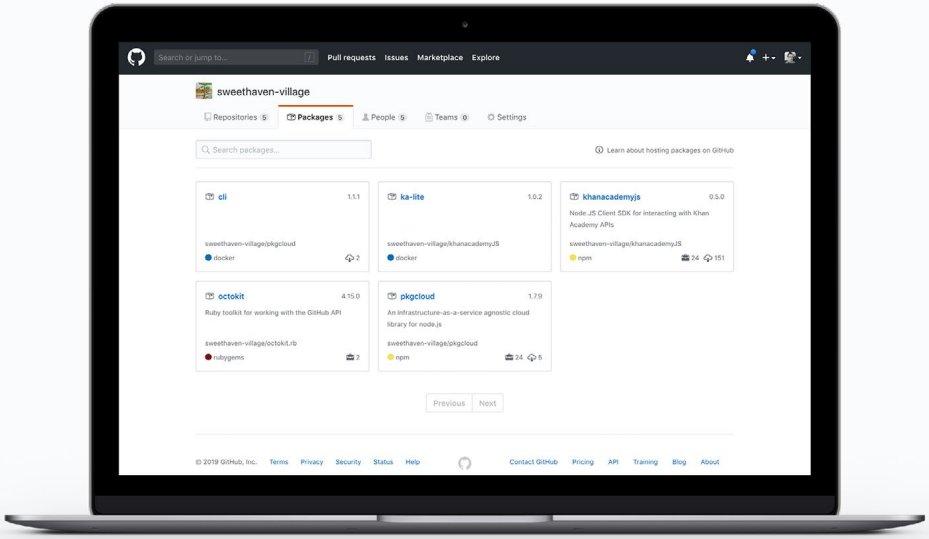
Ultimately, our goal is to better understand how security vulnerabilities start—and use this information to improve code vulnerability tools and identify issues earlier.

What's next

We plan to expand our traceability features, including Git history details on when the vulnerability was introduced, what reviews were associated with it, and more.

Packages

When you work on a project that has dependencies on packages, it's important to trust them, understand their code, and connect with the community who built them. And inside organizations, you need to be able to quickly find which packages have been approved to use. GitHub Packages makes it easy to use the same familiar GitHub interface to find public packages anywhere on GitHub, as well as private packages within your organization or repositories—securely. You can safely discover, use, and publish public and private packages in one place. Packages hosted on GitHub also include all the information you need—package contents, download statistics, version history, and more.



What's next

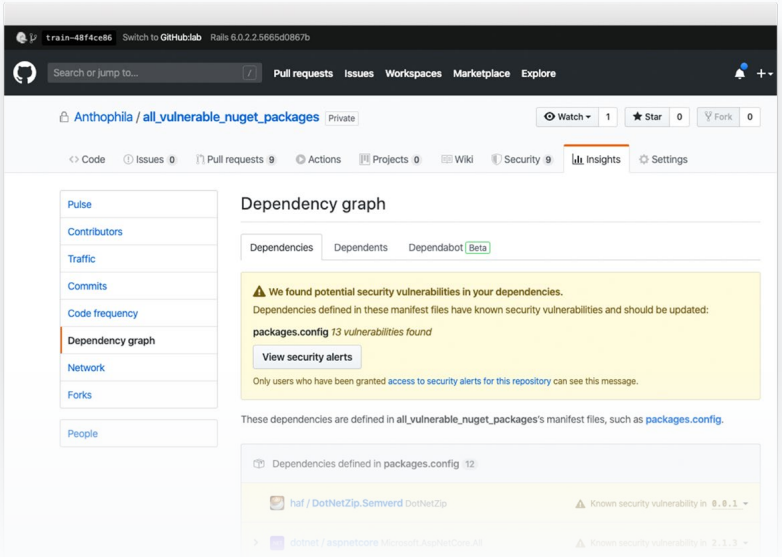
GitHub Packages supports familiar package management tools: JavaScript (npm), Java (Maven), Ruby (RubyGems), .NET (NuGet), and Docker images today, with many more to come.

Dependency insights

Security isn't just limited to individual repositories. Understanding your entire organization's security relies on much more, like understanding which languages you depend on so your team has the right staff to support

business-critical applications. Most of all, security relies on understanding how healthy your software is in real time.

GitHub gives organizations access to dependency insights, allowing teams to browse all software dependencies in use at one time, in one place. Dependency insights also make it easy to quickly filter and view dependencies with security advisories, or ones using a particular license—something especially critical for organizations with strict security and compliance policies. Finally, we've made it possible to list all the repositories and dependency manifests that include a particular dependency, so you can upgrade or remove dependencies that don't meet their requirements.



What's next

We'll continue to expand our dependency insights dashboard to include integrations and reporting along with other security tools.

Token scanning

Ensuring open source projects don't rely on vulnerable libraries is one way to make an immediate impact. Another is to help projects build and enforce secure coding practices and prevent security vulnerabilities before they are exposed. Repository access credentials are critical to keeping code secure, and are generally managed by teams.

Still, even the most secure organizations eventually make a mistake, and when leaked, access tokens become easy targets for attackers. Just as we've brought vulnerability and dependency security information natively to our platform, we want to help protect developers from leaking secrets as well.

The first step is GitHub token scanning—a scalable, real-time code scanning platform that we use to inspect incoming commits for sensitive information. Token scanning detects credentials from several platforms, including Amazon Web Services, Microsoft Azure, Google Cloud Platform, Slack, and Stripe. In public repositories, if a developer accidentally commits a credential to any of the supported services, we work with those services to identify the disclosure and proactively invalidate the credentials before malicious actors can use them in a compromising way. We've already identified millions of tokens since launching token scanning in 2018, and look forward to working with even more formats and tools going forward.

What's next

We'll help organizations build on what's currently available for public repositories. Organizations will be able to scan for tokens they define, including potential formats looking for personally identifiable information and other sensitive data.

Beyond these token formats, we'll continue to build on GitHub's code scanning platform, adding more features and integration points that can:

- Detect common security vulnerabilities in code as it's being written
- Detect more vulnerabilities, like username or password disclosures

And to support security features built by GitHub and the community, our Machine Learning and Data Science Teams will work through petabytes of data to better identify security issues. Knowing these will allow us to alert developers before vulnerable code gets merged and help them write more secure software from the start. Our goal is to stop open source projects from ever introducing security vulnerabilities, instead of only responding when they're found.

Activity insights

Open source projects are more than just their code. Like any organization, their popularity and impact ebb and flow over time. The most important toolchain today may

see its usage drop to near zero in just months. GitHub makes it easier for users to understand what's behind the code in each open source project.

With the Activity dashboard, organizations can have insights about the work their teams are doing. The dashboard provides information on development patterns, how your team is using the GitHub platform, dependency vulnerabilities, and GitHub issues. Teams can use this data to make better decisions and build more secure software by staying informed—like knowing the development languages used organization-wide and being aware of security vulnerabilities and how they're resolved.

Our **activity insights** also help organizations understand how their teams are using GitHub to collaborate and work on code. These insights also help you track, report, and act on your organization's open source usage.

Platform security, compliance, and health

To properly protect the code in GitHub projects, we need to make sure the platform itself is secure. Maintaining a software solution that services over 40 million users and thousands of businesses is a large task, especially when there are active efforts to try and cause disruption—including the largest DDOS attack yet recorded in early 2018.

Along with securing our platform, compliance is an important part of our security work. We're proud to say GitHub is SOC II compliant and has FedRAMP low certification.

Making it safe to build on open source code

As open source code gets more secure, so does the internal software that depends on it. While we use the same tools, features, and products to protect your private repositories, we also want to help your teams manage external code as well.

The first piece of this strategy is GitHub Connect: a set of features that brings tighter integration between GitHub Enterprise Cloud, our SaaS-based solution, and GitHub Enterprise Server, our self-hosted solution that can run on-premises, behind your firewall, or in a private cloud.

GitHub Connect

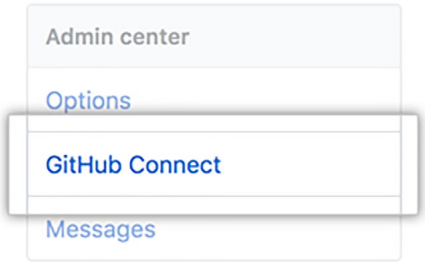
Secure open source is only helpful if you can easily use it within your own business. GitHub Connect lets you safely and securely connect to the world's largest community of software developers and open source projects on GitHub while keeping your most critical code protected behind the firewall. It also allows us to deliver features and data sourced from the public on GitHub to your business environment.

With unified search, developers can search open source and private repositories directly from within GitHub

Enterprise. Direct access to these repositories means you can leverage existing projects and better understand what your users are looking for—all within a more managed, visible, and secure workflow.

You can also take advantage of GitHub data—including critical security alerts and a clear path to vulnerability mitigation—through GitHub Connect. Developers and organizations are fixing vulnerabilities in their projects, but they don't necessarily notify others. With visibility into the aggregate data behind these fixes, we make sure you aren't using a vulnerable or outdated library—and we alert you if you do.

Finally, as a development leader you have many decisions to make, but GitHub Connect eliminates one important choice: staying on-premises or using the cloud. Connect to a community of innovation, maximize operational efficiencies, stay more secure, and provide unmatched developer experience—all while keeping your code as close as you need it to be.



What's next

These features are just the beginning. Next we'll integrate the fork and pull workflow used across open source projects today into GitHub Enterprise. Organizations will be able to fork code using a single encrypted gateway from GitHub into their enterprise environment, and have local access to it. You'll have access to the code, not just the released binaries, allowing you to scan and investigate with your own tools. And connected to GitHub, you can keep libraries up to date with just the push of a button, making it easy to integrate the latest security and critical bug fixes.

Further extending our searching capabilities, business administrators will have control over the code available by restricting the open source libraries users can bring into your environment based on license, reputation, or other factors. And by using the GitHub dependency graph, you can verify that all libraries in use came through approved channels.

Open source inside your organization isn't just about consuming the code; it's also about sharing code. Whether you're sharing your own projects with the world or upstreaming your bug fixes and patches to reduce overhead, GitHub Connect provides a connection from your secure environment to the open source projects on GitHub. More than just a

pull request, it will allow you to add additional checks to ensure that these open source patches or libraries have met your internal approvals, including legal, technical, and in some cases, upper management sign off as well. Building on the open source workflows that developers are used to—but with added review, approval, and compliance—can help organizations get the most out of open source.

GitHub as an extensible platform

GitHub has always had a best-of-breed philosophy. We're building a platform that allows our partners to create seamless integrations and extend GitHub with new features, functionalities, and workflows. This strategy also holds true for security tools.

We've seen different organizations take different strategies with their security workflows. Some businesses integrate the one tool that best meets their needs, while others integrate multiple tools with the idea that breach prevention is worth any amount of money spent. With the introduction of GitHub Actions, it's easier than ever to integrate the tools you need.

Whatever your strategy is, you likely don't want to leave security to chance. Being able to integrate tools from leaders in the security space—including Black Duck, HP, IonChannel, Sonatype, Snyk, and White Source—ensures you're able to use the latest applications and services to keep your business secure. And when new tools come along, you can update or replace existing tools as easily as you added them.

If you use in-house tools to manage threat intelligence or identify software vulnerabilities, you can also integrate those into our platform using our APIs or replace them as necessary. Creating your own alerts within our Advisory Database or using our dependency APIs to better understand the libraries and code already in use can help you shift security left—integrating and benefiting from your strategies earlier in your software lifecycle.

What's next

GitHub's platform strategy is only accelerating, with additional integrations, partners adding support for GitHub, and new data sets to make our partners' interactions even richer. Just like how the best software is built by a diverse team of developers, the best software is developed by a diverse set of tools. At GitHub, we want to make this easier and more powerful.

Questions about using open source in your organization? We're here to help.

SALES@GITHUB.COM
GITHUB.COM/ENTERPRISE

GitHub