About repository security advisories

In this article

You can use repository security advisories to privately discuss, fix, and publish information about security vulnerabilities in your repository.

Anyone with admin permissions to a repository can create a security advisory.

Anyone with admin permissions to a repository also has admin permissions to all security advisories in that repository. People with admin permissions to a security advisory can add collaborators, and collaborators have write permissions to the security advisory.

> **Note:** If you are a security researcher, you should directly contact maintainers to ask them to create security advisories or issue CVEs on your behalf in repositories that you don't administer. However, if private vulnerabiliy reporting is enabled for the repository, you can *privately* report a vulnerability yourself. For more information, see "Privately reporting a security vulnerability."

About repository security advisories

Vulnerability disclosure is an area where collaboration between vulnerability reporters, such as security researchers, and project maintainers is very important. Both parties need to work together from the moment a potentially harmful security vulnerability is found, right until a vulnerability is disclosed to the world, ideally with a patch available. Typically, when someone lets a maintainer know privately about a security vulnerability, the maintainer develops a fix, validates it, and notifies the users of the project or package. For more information, see "About coordinated disclosure of security vulnerabilities."

Repository security advisories allow repository maintainers to privately discuss and fix a security vulnerability in a project. After collaborating on a fix, repository maintainers can publish the security advisory to publicly disclose the security vulnerability to the project's community. By publishing security advisories, repository maintainers make it easier for their community to update package dependencies and research the impact of the security vulnerabilities.

With repository security advisories, you can:

1. Create a draft security advisory, and use the draft to privately discuss the impact of the vulnerability on your project. For more information, see "Creating a repository security advisory."
2. Privately collaborate to fix the vulnerability in a temporary private fork.
3. Publish the security advisory to alert your community of the vulnerability once a patch is released. For more information, see "Publishing a repository security

advisory."

You can also use repository security advisories to republish the details of a security vulnerability that you have already disclosed elsewhere by copying and pasting the details of the vulnerability into a new security advisory.

You can also use the REST API to create, list, and update repository security advisories. For more information, see "Repository security advisories" in the REST API documentation.

You can give credit to individuals who contributed to a security advisory. For more information, see "Editing a repository security advisory."

You can create a security policy to give people instructions for reporting security vulnerabilities in your project. For more information, see "Adding a security policy to your repository."

If you created a security advisory in your repository, the security advisory will stay in your repository. We publish security advisories for any of the ecosystems supported by the dependency graph to the GitHub Advisory Database on github.com/advisories. Anyone can submit a change to an advisory published in the GitHub Advisory Database. For more information, see "Editing security advisories in the GitHub Advisory Database."

If a security advisory is specifically for npm, we also publish the advisory to the npm security advisories. For more information, see npmjs.com/advisories.

You can also join GitHub Security Lab to browse security-related topics and contribute to security tools and projects.

CVE identification numbers

GitHub Security Advisories builds upon the foundation of the Common Vulnerabilities and Exposures (CVE) list. The security advisory form on GitHub is a standardized form that matches the CVE description format.

GitHub is a CVE Numbering Authority (CNA) and is authorized to assign CVE identification numbers. For more information, see "About CVE" and "CVE Numbering Authorities" on the CVE website.

When you create a security advisory for a public repository on GitHub, you have the option of providing an existing CVE identification number for the security vulnerability. If you want a CVE identification number for the security vulnerability in your project, and don't already have one, you can request a CVE identification number from GitHub. GitHub usually reviews the request within 72 hours. Requesting a CVE identification number doesn't make your security advisory public. If your security advisory is eligible for a CVE, GitHub will reserve a CVE identification number for your advisory. We'll then publish the CVE details after you make your security advisory public. Anyone with admin permissions to a security advisory can request a CVE identification number.

If you already have a CVE you want to use, for example, if you use a CVE Numbering Authority (CNA) other than GitHub, add the CVE to the security advisory form. This may happen, for example, if you want to get the advisory consistent with other communications you plan to send out at publication time. GitHub cannot assign CVEs to your project if it is covered by another CNA.

Once you've published the security advisory and GitHub has assigned a CVE identification number to the vulnerability, GitHub publishes the CVE to the MITRE database. For more information, see "Publishing a repository security advisory."

Dependabot alerts for published security advisories

GitHub will review each published security advisory, add it to the GitHub Advisory Database, and may use the security advisory to send Dependabot alerts to affected repositories. If the security advisory comes from a fork, we'll only send an alert if the fork owns a package, published under a unique name, on a public package registry. This process can take up to 72 hours and GitHub may contact you for more information.

For more information about Dependabot alerts, see "About Dependabot alerts" and "About Dependabot security updates." For more information about GitHub Advisory Database, see "Browsing security advisories in the GitHub Advisory Database."

Legal