

GitHub Advisory Database now powers npm audit

Today, we're adding a proxy on top of the GitHub Advisory Database that speaks the `npm audit` protocol. This means that every version of the npm CLI that supports security audits is now talking directly to the GitHub Advisory Database.



By Edward Thomson

October 7, 2021



Supply chain security is one of the most important parts of software development today, and we want to make developing securely as easy as possible for developers. Today, we're taking another step in bringing all this together for both npm and GitHub by announcing that the GitHub Advisory Database now powers `npm audit`.

`npm audit` is a command that you can run in your Node.js application to scan your project's dependencies for known security vulnerabilities—you'll be given a URL that you can visit to learn more, and information about what versions have fixed this vulnerability. In addition, the `npm install` command uses this information to give you a brief summary of problems.

```
up to date, audited 252 packages in 1s
```

```
6 vulnerabilities (3 low, 1 moderate, 2 high)
```

```
To address issues that do not require attention, run:
npm audit fix
```

Integrating npm's security systems

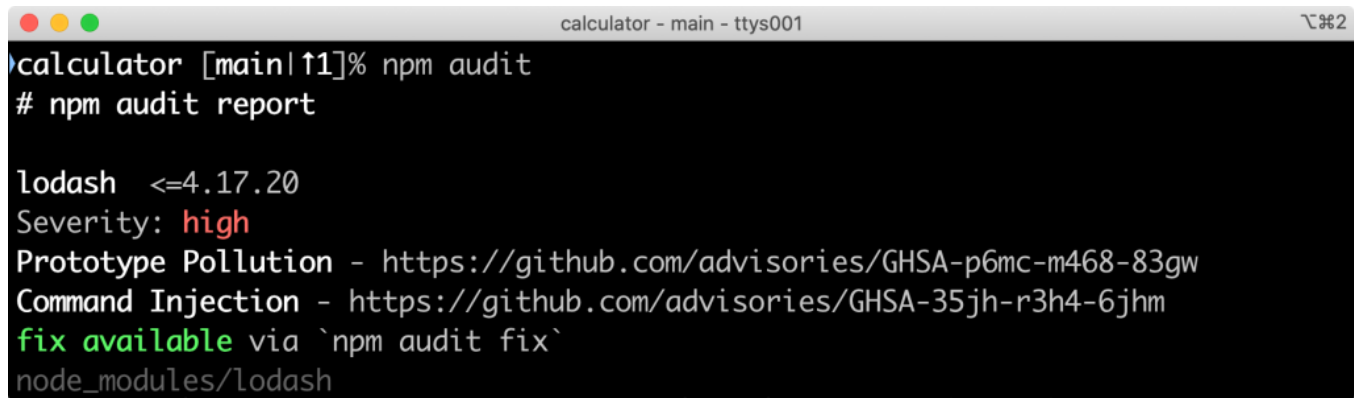
The GitHub Advisory Database is a carefully curated set of more than 5,000 security vulnerabilities that powers important security tools like Dependabot. When npm joined GitHub, the npm advisory database became a part of our portfolio of security products, but

(unfortunately) that meant that we had two databases of security advisories.

Last year, we added all the npm security advisories to the GitHub Advisory Database. By doing this, we made sure that you were seeing the same advisories for your project—whether you were scanning it with `npm audit` or a tool like Dependabot. This was a great first step because developers didn't have to look in two places to see security advisories for their dependencies, but for GitHub we still had differences between the schemas in each database. This made it harder to add new features, and also created extra work since our security engineers who curate these advisories needed to make sure that each advisory was accurate in each database.

GitHub Advisory Database + npm

Today, we're adding a proxy on top of the GitHub Advisory Database that speaks the `npm audit` protocol. This means that every version of the npm CLI that supports security audits is now talking directly to the GitHub Advisory Database.

A terminal window titled 'calculator - main - ttys001' shows the output of the 'npm audit' command. The output indicates a high severity vulnerability in 'lodash' (version <=4.17.20) due to 'Prototype Pollution' and 'Command Injection'. It provides links to the GitHub advisories (GHSA-p6mc-m468-83gw and GHSA-35jh-r3h4-6jhm) and states that a fix is available via 'npm audit fix'. The path 'node_modules/lodash' is also shown.

```
calculator [main|11]% npm audit
# npm audit report

lodash <=4.17.20
Severity: high
Prototype Pollution - https://github.com/advisories/GHSA-p6mc-m468-83gw
Command Injection - https://github.com/advisories/GHSA-35jh-r3h4-6jhm
fix available via `npm audit fix`
node_modules/lodash
```

In addition, we're redirecting the advisories on npmjs.com to the GitHub Advisory Database. This means you can view advisories and also search and sort advisories in a more advanced way. Every developer gets the same, high-quality vulnerability information from the GitHub Advisory Database, and we'll stay focused on keeping developing on npm and GitHub secure.

Learn more

Jump in and explore npm advisories today, or learn more about our other supply chain security features as follows:

- `npm audit`
- The Advisory Database
- Security advisories
- Dependency graph
- Dependabot alerts
- Dependabot security updates