

Forum:[Hack Forums](#)

Thread: [NPM audit fix](#)

bevc - May 18, 2019, 10:17 pm

I'm just curious about this. This morning I installed a package using npm, and I got a warning about a vulnerability. I ran the 'npm audit' command and it gave me some information about a vulnerability that has been detected in one of the previously installed packages.

It says that to fix this vulnerability, all you need to do is run 'npm audit fix'. Sounds a little too good to be true, so I googled around, and did find at least one article advising against doing this, for the reason that immediately occurred to me: you might install the 'fix' and suddenly a load of the functionality you needed has been deprecated/disappeared and the application crashes like a Michael Bay movie.

I had a google for this 'vulnerability' as well but couldn't find much information about it.

This is more just a speculative question, does anyone have any thoughts on 'npm audit fix'? Has anyone used it and had it break something? I'm always wary of this kind of thing.

Dave85 - May 19, 2019, 1:51 am

I have never using it.

I already tried once but it resulted that I should resolve it manually.

So I always fix these manually.

Some module have already a patch for the vulnerability, for example:

Code === npm audit security report ===

Run npm update fstream --depth 5 to resolve 2 vulnerabilities

But some module dont have an update.

But there should be someone which have same issue and opened an issue on the module repo (github) and I would just follow the issue, or just wait for an update.

bevc - May 19, 2019, 4:13 am

(May 19, 2019 01:51 AM)Dave85 Wrote: [\[CITE\]](#) Some module have already a patch for the vulnerability, for example:

Code === npm audit security report ===

Run npm update fstream --depth 5 to resolve 2 vulnerabilities

But some module dont have an update.

But there should be someone which have same issue and opened an issue on the module repo (github) and I would just follow the issue, or just wait for an update.

Yeah, but what if the update overwrites some functionality that you need which is now deprecated? This is always my worry with this kind of thing, that I'll run this update and everything will seem fine, then on Monday morning I'll get a phone call saying that something has broken. So I ask myself whether it's worth it.

Foch - August 5, 2019, 4:06 pm

(May 19, 2019 04:13 AM)bevc Wrote: [\[CITE\]](#) Yeah, but what if the update overwrites some functionality that you need which is now deprecated? This is always my worry with this kind of thing, that I'll run this update and everything will seem fine, then on Monday morning I'll get a phone call saying that something has broken. So I ask myself whether it's worth it.

You got to act case-by-case anyway. npm audit fix should be ok most of the time. Actually, it never fucked up any of my apps but I have to admit I rarely encounter it.

Anyway, if the update overwrites fonctionnalités, just delete node_modules and rerun npm install.

vorticalbox - September 5, 2019, 2:59 am

(May 19, 2019 04:13 AM)bevc Wrote: [\[CITE\]](#) Yeah, but what if the update overwrites some functionality that you need which is now deprecated? This is always my worry with this kind of thing, that I'll run this update and everything will seem fine, then on Monday morning I'll get a phone call saying that something has broken. So I ask myself whether it's worth it.

npm audit won't version bump to majors only minor and patch. E.G if there is a vulnerability in 1.5.6 and the fix is in 1.6.0 then audit will bump the version for you (there shouldn't be any API changes assuming people follow <https://semver.org/>) but it will not bump if the fix is in version 2.0.0

Anzee - September 5, 2019, 8:42 am

Anything you can install via a package manager you can build and install from source.

Check if the newer version you're updating to contains all necessary functionality and update to it or check if the vulnerability even affects your production environment.

I could care less about something like authenticated denial of service in an intranet environment than unauthenticated RCE on a public facing servlet.