# Checando suas dependências

Você confia nas dependências das suas dependências?

## Inajara Leppa

Consultora de Desenvolvimento na ThoughtWorks e representante do pilar de Justiça Econômica e Social em Porto Alegre

### **Thought**Works<sup>®</sup>







### Lucas Vieira

Consultor de Desenvolvimento na ThoughtWorks

consultor de Descrivorviniento na modgittworks



**Thought**Works<sup>®</sup>

### **OWASP TOP 10**

A9:2017: Utilizando componentes com vulnerabilidades conhecidas

#### Sua aplicação está vulnerável?

- Se você não souber as versões de todos os componentes usados. Isso inclui componentes que você usa diretamente, bem como dependências aninhadas.
- Se você não verificar vulnerabilidades regularmente e se inscrever em boletins de segurança relacionados aos componentes que você usa.
- Se você não corrigir ou atualizar a plataforma, as estruturas e as dependências subjacentes de maneira oportuna e baseada em risco.
- Se os desenvolvedores de software não testarem a compatibilidade de bibliotecas atualizadas, atualizadas ou corrigidas.

## "Security is always excessive until it is not enough."

- Robbie Sinclair

Head of Security, Country Energy, NSW Australia



### O que é?

- Comando que analisa as dependências da sua aplicação, buscando vulnerabilidades conhecidas pela comunidade.
- npm audit checa dependencias diretas, devDependencies, bundledDependencies, e optionalDependencies, mas não checa peerDependencies.
- Está disponível a partir da versão 6 do npm

### Como faz?

- > npm audit [--json|--parseable]
- --json Relatório detalhado em formato JSON
- --parseable Relatório detalhado em formato de texto

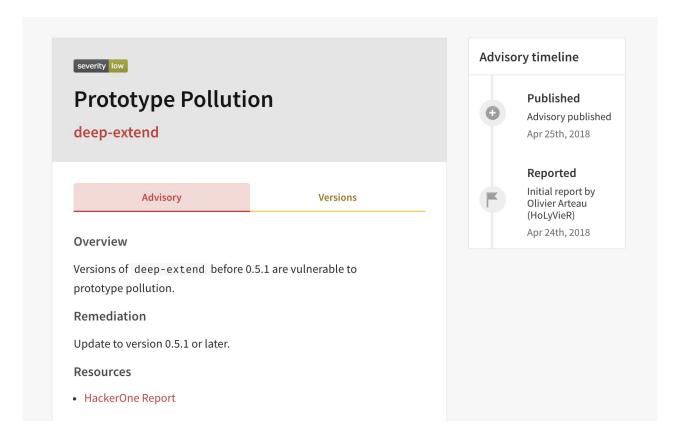
Manual Review
Some vulnerabilities require your attention to resolve
Visit https://go.npm.me/audit-guide for additional guidance

Moderate	Prototype pollution	
Package	hoek	
Patched in	> 4.2.0 < 5.0.0    >= 5.0.3	
Dependency of	numbat-emitter	
Path	numbat-emitter > request > hawk > boom > hoek	
More info	https://nodesecurity.io/advisories/566	

=== npm audit security report ===			
# Run npm install chokidar@2.0.3 to resolve 1 vulnerability SEMVER WARNING: Recommended action is a potentially breaking change			
Low	Prototype Pollution		
Package	deep-extend		
Dependency of	chokidar		
Path	chokidar > fsevents > node-pre-gyp > rc > deep-extend		
More info	https://nodesecurity.io/advisories/612		

Vulnerabilidade que necessita de revisão manual

Vulnerabilidade com ação recomendada para resolução



©ThoughtWorks 2018 Commercial in Confidence

Severidade	Ação Recomendada
Crítica (Critical)	Endereçar imediatamente
Alta (High)	Endereçar o quanto antes
Moderada (Moderate)	Endereçar quando possível
Baixa (Low)	Endereçar sem pressa

É possível passar o nível mínimo de severidade no npm config (audit-level)



### O que é?

• Comando que instala automaticamente as atualizações para as dependências vulneráveis que não necessitam de revisão manual.

### Como faz?

```
> npm audit fix [--only|--dry-run]
```

- --only=prod executa apenas nas dependências diretas
- --dry-run mostra prévia do que vai acontecer ao executar o comando



• Se o projeto utiliza um registry privado para o npm (ex: jfrog artifactory), é necessário passar o registry público junto do comando e o audit será executado apenas nos repositórios públicos do npm.

ex: npm audit --registry=https://registry.npmjs.org

 Não recomendamos utilizar o comando de fix caso sua aplicação utilize um registry privado, pois isso irá alterar a url da dependência para apontar para o registry público. A solução para isso é executar os comandos recomendados pelo audit manualmente. • Diferente do comando de fix, o comando de audit ainda não suporta verificar apenas dependências de produção e essa verificação deve ser feita manualmente.

• Como forma de trazer esse olhar de segurança para o time, adicionamos essa verificação em nossa pipeline, que é executada 2x por dia e caso encontre vulnerabilidades envia uma mensagem em nosso canal de alertas do time.



- <a href="https://github.com/thoughtworks/talisman">https://github.com/thoughtworks/talisman</a>: A tool to detect and prevent secrets from getting checked in.
- <a href="https://github.com/AGWA/git-crypt">https://github.com/AGWA/git-crypt</a>: git-crypt enables transparent encryption and decryption of files in a git repository.
- <a href="https://github.com/zaproxy/zaproxy">https://github.com/zaproxy/zaproxy</a>: It can help you automatically find security vulnerabilities in your web applications while you are developing and testing your applications. It's also a great tool for experienced pentesters to use for manual security testing.
- <a href="https://yarnpkg.com/lang/en/docs/cli/audit/">https://yarnpkg.com/lang/en/docs/cli/audit/</a>: Yarn also has an audit tool, but we didn't test it as much as the npm one.

## Obrigada

Inajara Leppa ileppa@thoughtworks.com Lucas Vieira lucas.f@thoughtworks.com