



Mahidol University
Faculty of Information
and Communication Technology



SEE TO BELIEVE: Using Visualization to Motivate Updating Third-party Dependencies



C. Ragkhitwetsagul, V. Jarukitpipat, M. Choetkiertikul, K. Chhun, W. Wanprasert, T. Sunetnanta, Faculty of ICT, Mahidol University, Thailand

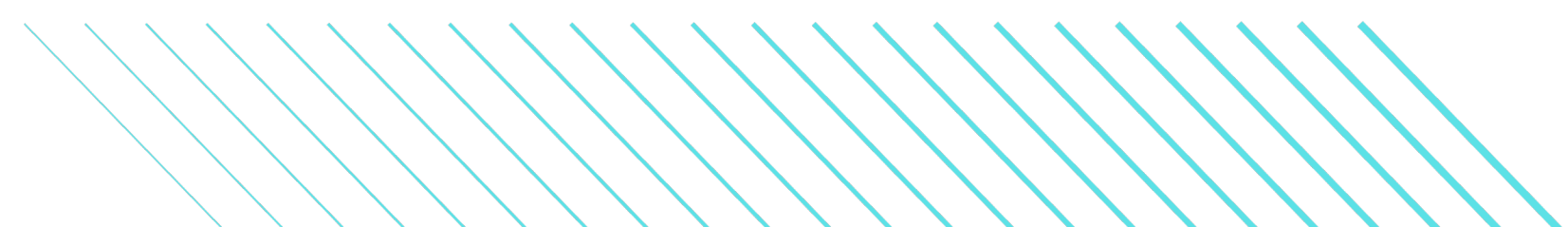
Raula Gaikovina Kula, NAIST, Japan



Software Engineering Laboratory
Graduate School of Information Science
Nara Institute of Science and Technology



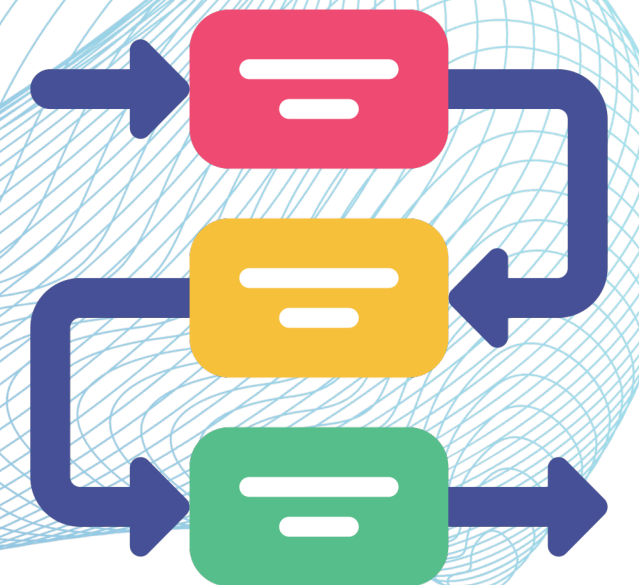
JCSSE 2024, 19-21 July 2024



THIRD-PARTY DEPENDENCIES

A **dependency** is additional code that a programmer wants to call.

Adding a dependency avoids repeating work already done: designing, writing, testing, debugging, and maintaining a specific unit of code.





master

mocha / package.json

Code

Blame

171 lines (171 loc) · 4.44 KB

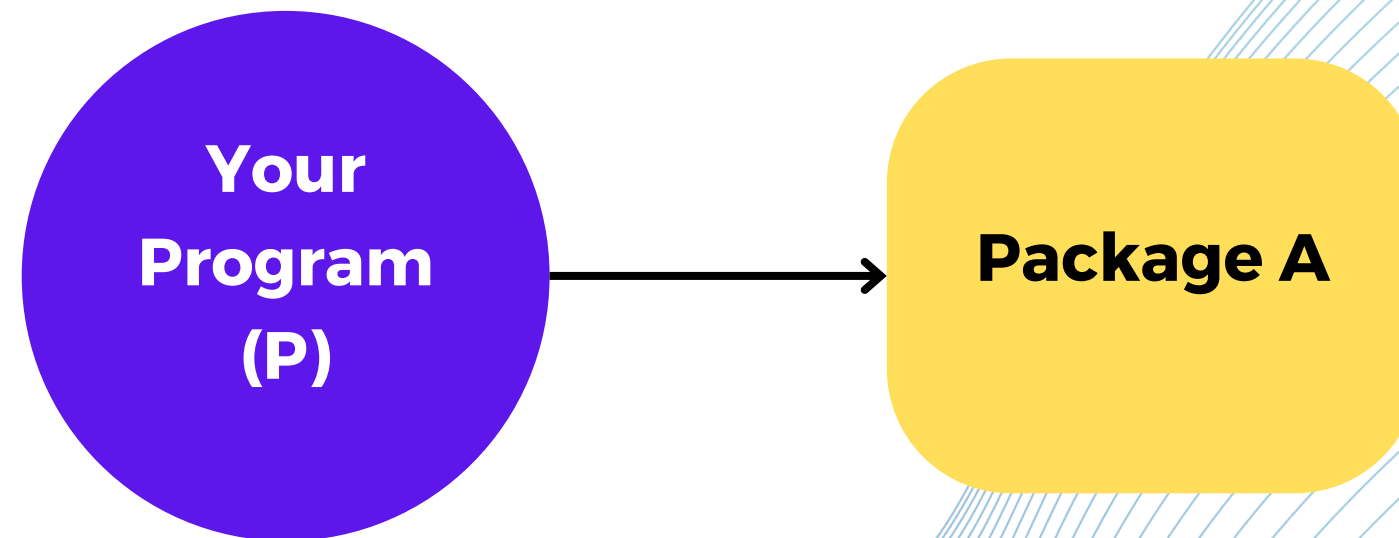
```
52     },
53     "dependencies": {
54       "ansi-colors": "4.1.1",
55       "browser-stdout": "1.3.1",
56       "chokidar": "^3.5.3",
57       "debug": "4.3.4",
58       "diff": "5.0.0",
59       "escape-string-regexp": "4.0.0",
60       "find-up": "5.0.0",
61       "glob": "8.1.0",
62       "he": "1.2.0",
63       "js-yaml": "4.1.0",
64       "log-symbols": "4.1.0",
65       "minimatch": "5.0.1",
66       "ms": "2.1.3",
67       "serialize-javascript": "6.0.0",
68       "strip-json-comments": "3.1.1",
69       "supports-color": "8.1.1",
70       "workerpool": "6.2.1",
71       "yargs": "16.2.0",
72       "yargs-parser": "20.2.4",
73       "yargs-unparser": "2.0.0"
74     },
```



TWO TYPES OF DEPENDENCIES

Direct dependency:

$(P \rightarrow A)$



Transitive dependency:

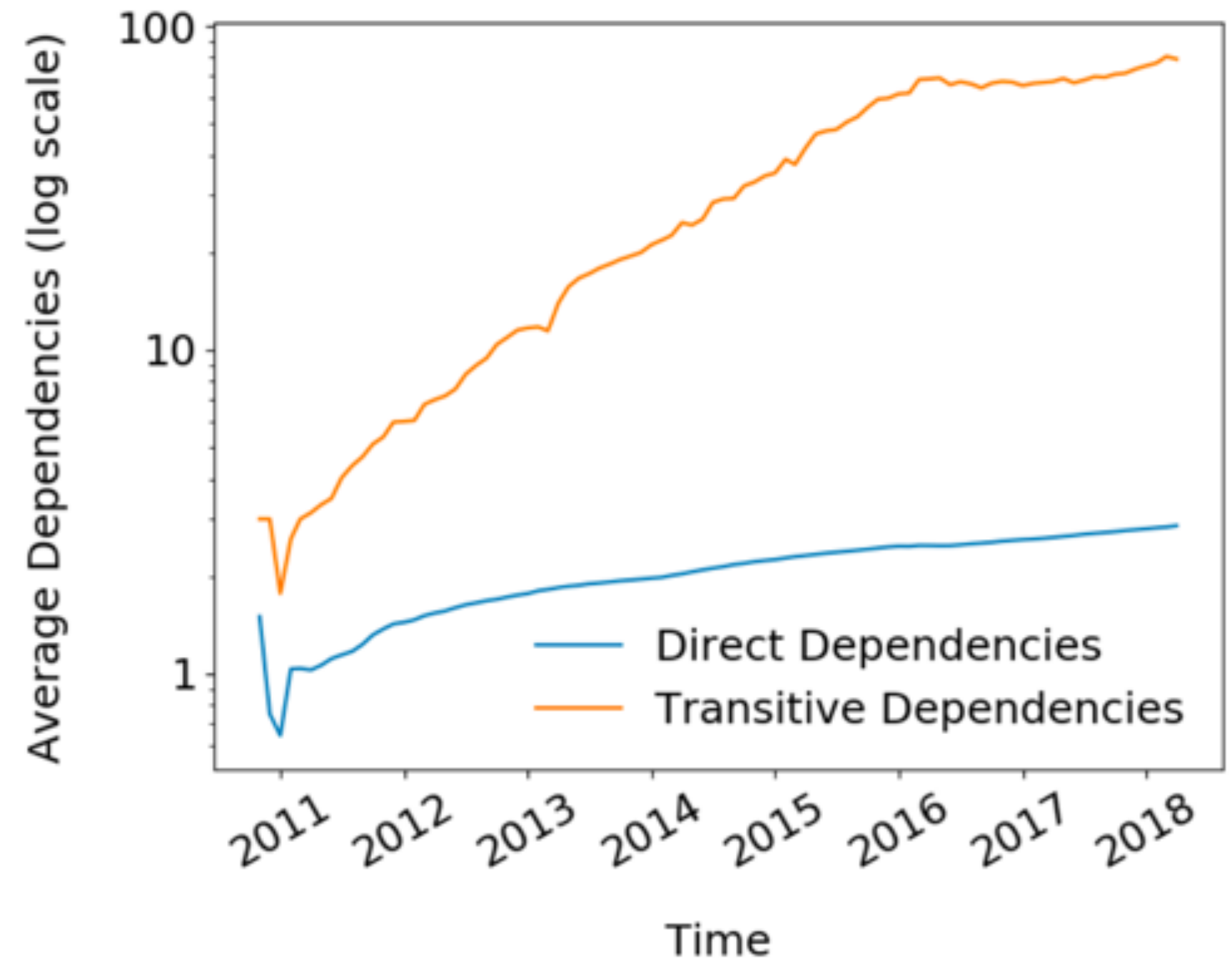
$(P \rightarrow A \rightarrow B)$



NPM ECOSYSTEM

Node.js dependency manager **NPM (Node Package Manager)** provides access to more than 750,000 packages

Number of **transitive dependencies per one direct dependency is 80.**



Zimmermann et al., (2019). Small world with high risks: A study of security threats in the NPM ecosystem. Proceedings of the 28th USENIX Security Symposium, 995-1010.

DEPENDENCY VULNERABILITIES

The usage of third-party dependencies may lead to security vulnerabilities.

GitHub Advisory Database (<https://github.com/advisories>) contains a curated list of security vulnerabilities



GITHUB ADVISORY DATABASE

GitHub

Q Type [7] to search

GitHub Advisory Database

Security vulnerability database inclusive of CVEs and GitHub originated security advisories from the world of open source software.

GitHub reviewed advisories



- All reviewed 19,314
- Composer 3,956
- Erlang 29
- GitHub Actions 16
- Go 1,740
- Maven 4,967
- npm 3,507
- NuGet 609
- pip 3,064
- Pub 10
- RubyGems 832
- Rust 780
- Swift 34

Unreviewed advisories

Q Search by CVE/GHSA ID, package, severity, ecosystem, credit...

19,314 advisories

Severity ▾ CWE ▾ Sort ▾

- TinyMCE Cross-Site Scripting (XSS) vulnerability using noneditable_regexp option** Moderate
CVE-2024-38356 was published for TinyMCE (Composer) 15 hours ago
- TinyMCE Cross-Site Scripting (XSS) vulnerability using noscript elements** Moderate 
- socket.io has an unhandled 'error' event** High 
- curve25519-dalek has timing variability in `curve25519-dalek`'s `Scalar29::sub`/`Scalar52::sub`** Moderate
- Moodle CSRF risks due to misuse of confirm_sesskey** Moderate
- Moodle HTTP authorization header is preserved between "emulated redirects"** Moderate
- Moodle BigBlueButton web service leaks meeting joining information** Moderate



EXISTING TOOL SUPPORT

DEPENDABOT

The screenshot shows a GitHub pull request interface. At the top, navigation tabs include Code, Issues, Pull requests (2), Actions, Projects, Wiki, Security (10), Insights, and Settings. The left sidebar contains sections for Overview, Reporting, Policy, Advisories, Vulnerability alerts, and Dependabot (10). The main content area displays the pull request title 'Bump nokogiri from 1.13.3 to 1.13.9 #5' with an 'Open' status. A blue warning box states: 'Merging this pull request will resolve 7 Dependabot alerts on nokogiri including a high severity alert.' Below this, a conversation from the 'dependabot' bot is shown, dated Oct 21, 2022. The comment reads: 'Bumps nokogiri from 1.13.3 to 1.13.9.' It includes expandable sections for 'Release notes', 'Changelog', and 'Commits'. A 'compatibility 82%' badge is visible. The comment also states: 'Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting @dependabot rebase.' The right sidebar shows 'Reviewers' (cragkhit), 'Assignees' (None), 'Labels' (dependencies), and 'Projects' (None yet). At the bottom, the commit 'Bump nokogiri from 1.13.3 to 1.13.9' is shown as 'Verified' with hash dce04b8.

NPM AUDIT

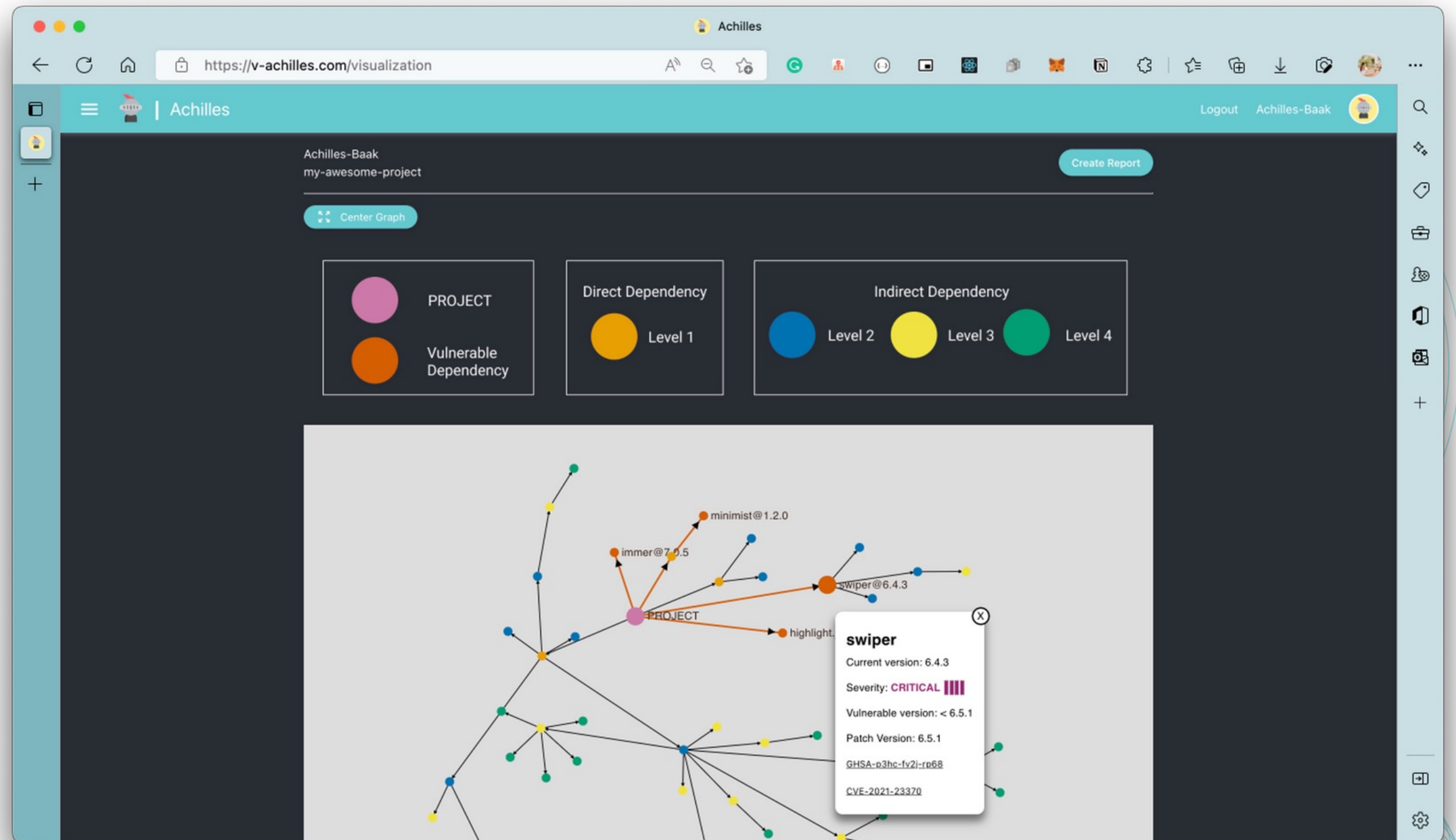
```
=== npm audit security report ===
```

```
# 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

We posit that, given a visual representation, the developers may re-prioritize their decisions to update the dependencies.

V-ACHILLES











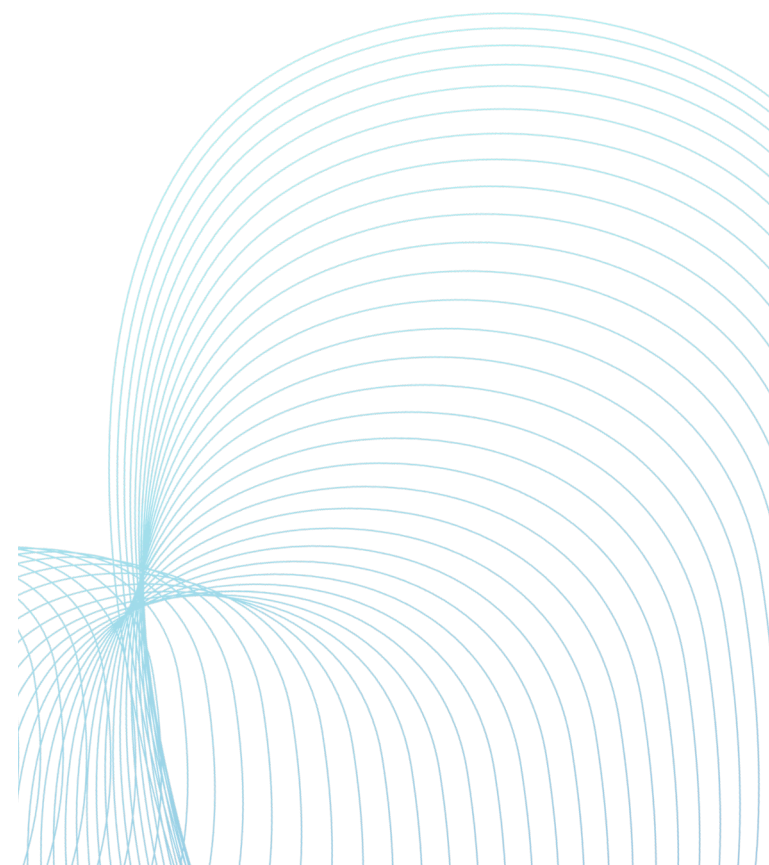
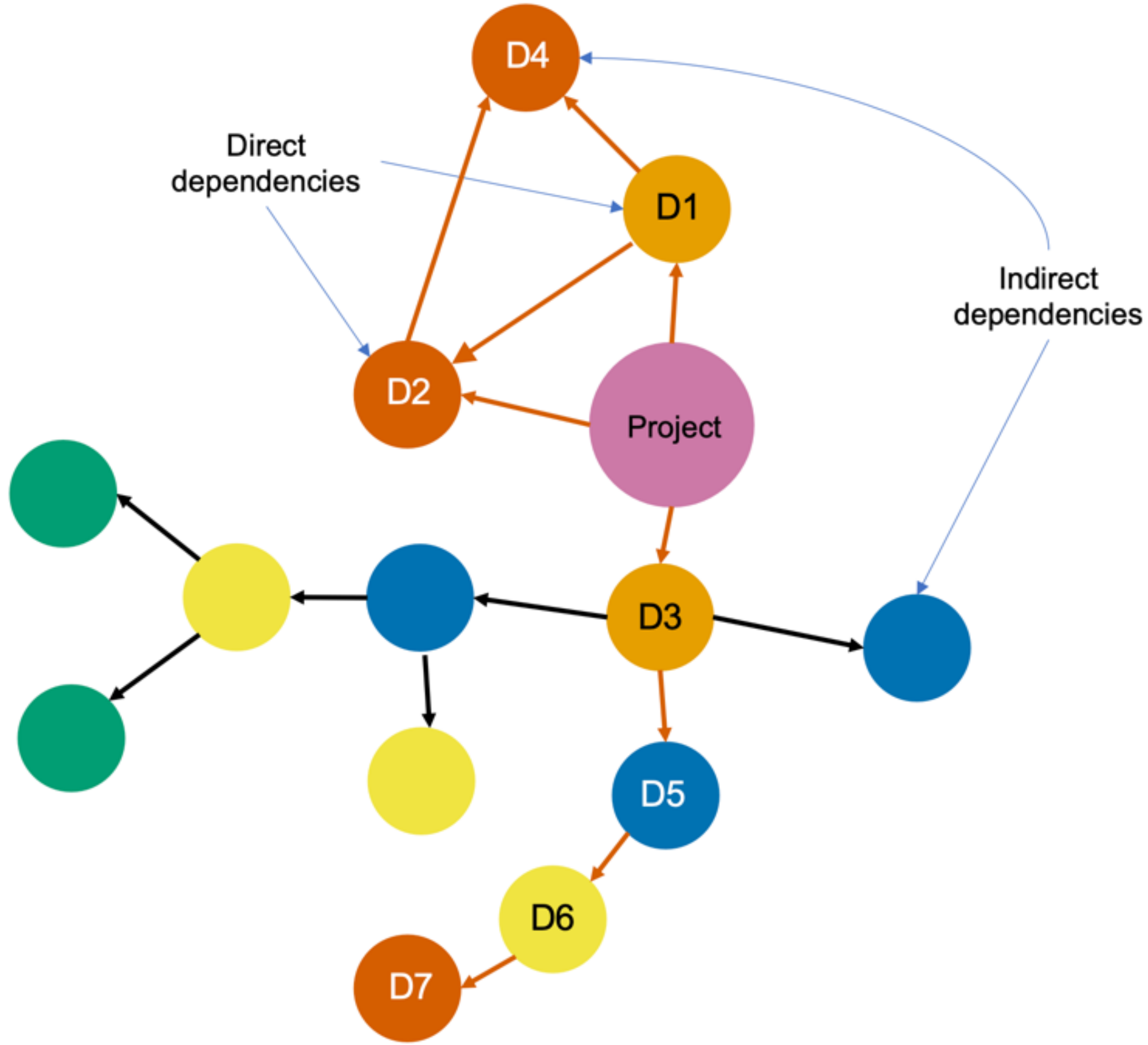
Jarukitpipat, V., Chhun, K., Wanprasert, W., Choetkiertikul, M., Sunetnanta, T., Kula, R. G., Chinthanet, B., Ishio, T., & Matsumoto, K. (2022).

V-Achilles: An Interactive Visualization of Transitive Security Vulnerabilities.

The 37th IEEE/ACM International Conference on Automated Software Engineering (ASE).

DEPENDENCY GRAPH VISUALIZATION

 Project
 Vulnerable Dependency
Direct Dependency
 Direct dependency
Indirect Dependency
 Level-1 indirect dependency
 Level-2 indirect dependency
 Level-3 indirect dependency
Dependency Link
 Normal dependency path
 Vulnerable dependency path



Which GitHub repository do you want to find vulnerabilities?

Search repositories by name...

Personal Repositories

- 2021-IST-Achilles
- achilles-react
- campaign-critic
- covid-dashboard
- cv
- ESCheckerM
- FileConverters
- GitHub-Crawler
- iwsc2018
- MethodExtractor
- mozanalysis
- MyLife
- Raandee
- SimCal
- StackoverflowChecker
- achilles-bootstrap
- appengine-autotweeter
- Chips-n-Salsa
- cragkhit.github.io
- deeplearning4j
- es_exp
- GACloneAgreement
- hello-github-actions
- JavaTokenizer
- MethodParser
- musicg
- onlineclone_processor
- Rational
- sortingalgo
- Thai-IT-community
- achilles-demo
- CalAgreedLOC
- cloverflow-web
- crjk-iwsc17
- demo-for-achilles
- evoESParamSearch
- ghtorrent.org
- hijack
- jpacman
- MinHashCloneDetector
- my-awesome-project
- r-community-explorer
- SiameseX
- ssbsechallenge2016
- the_facebook_scandal



RESEARCH QUESTION

To what extent does our visualization influence the developer's decision to update?

EMPIRICAL STUDY

TASKS

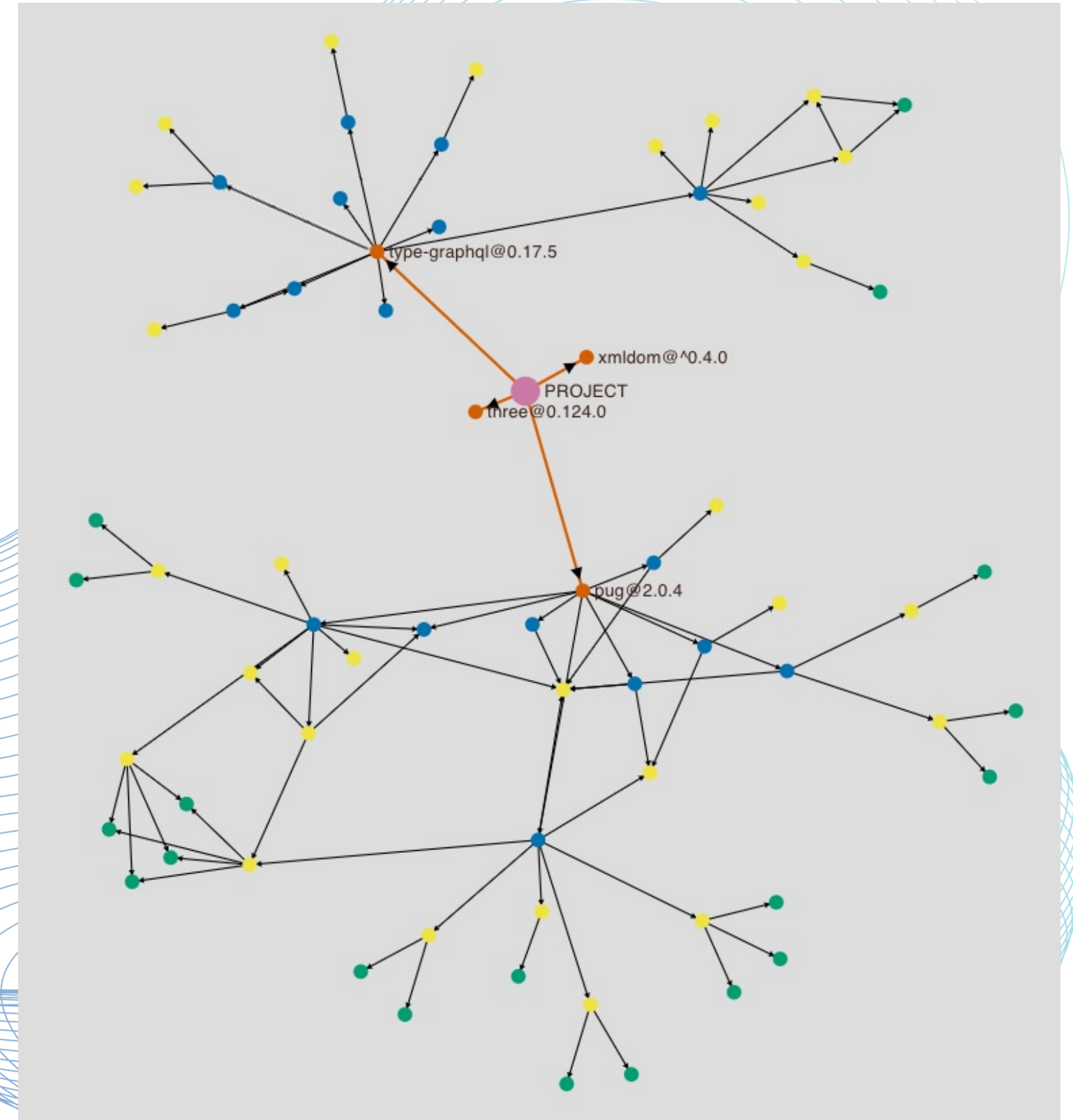
We compare V-Achilles to Dependabot and npm audit, using two tasks

Task 1: Navigating dependencies with complex graphs

Task 2: Navigating transitive dependencies with vulnerabilities

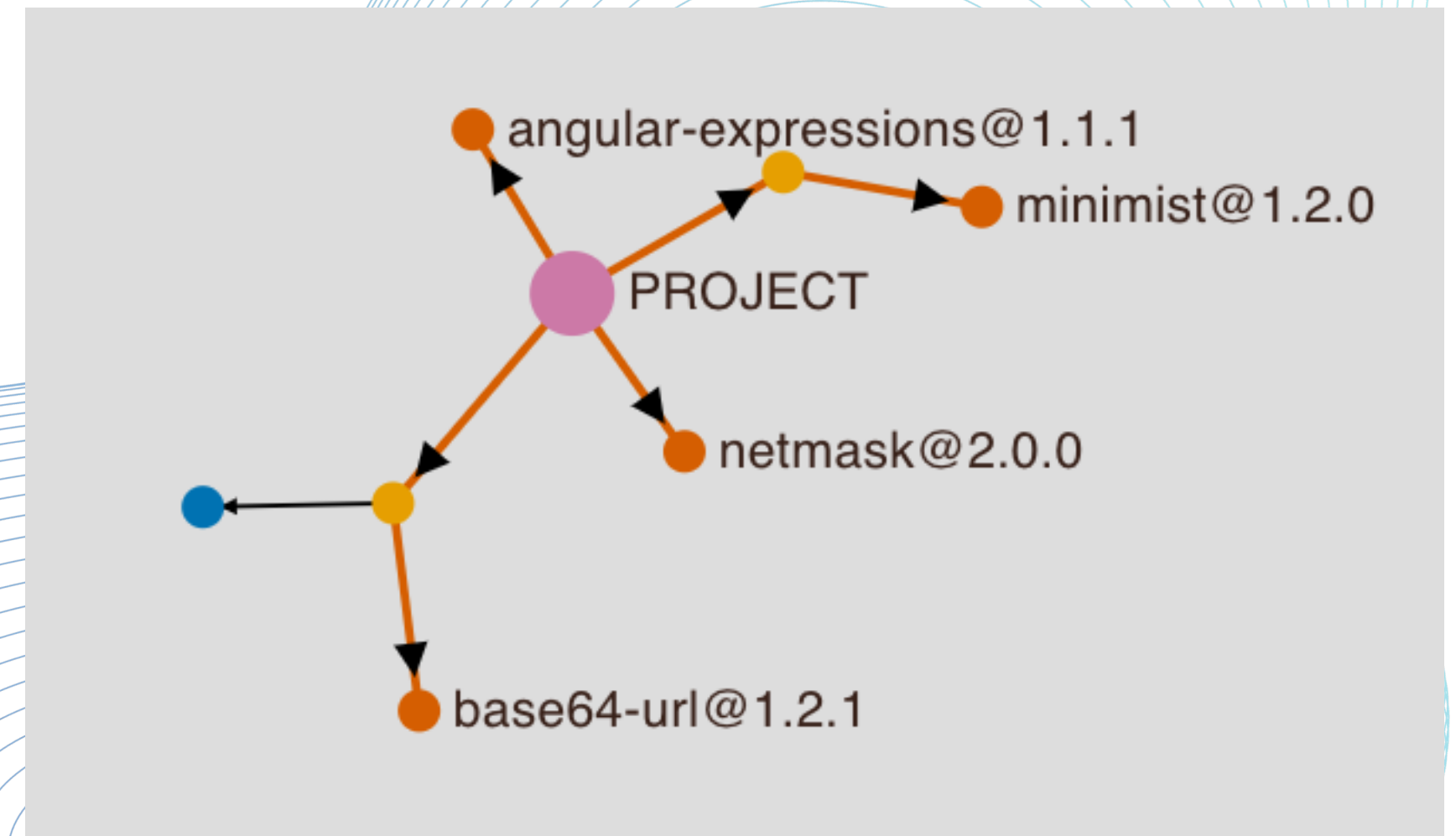
TASK 1: NAVIGATING DEPENDENCIES WITH COMPLEX GRAPHS

No	Dependency	Version	Severity	Type
1	three	0.124.0	High	Simple
2	pug	2.0.4	High	Complex
3	xmldom	^0.4.0	Low	Simple
4	type-graphql	0.17.5	Low	Complex



TASK 2: NAVIGATING TRANSITIVE DEPENDENCIES WITH VULNERABILITIES

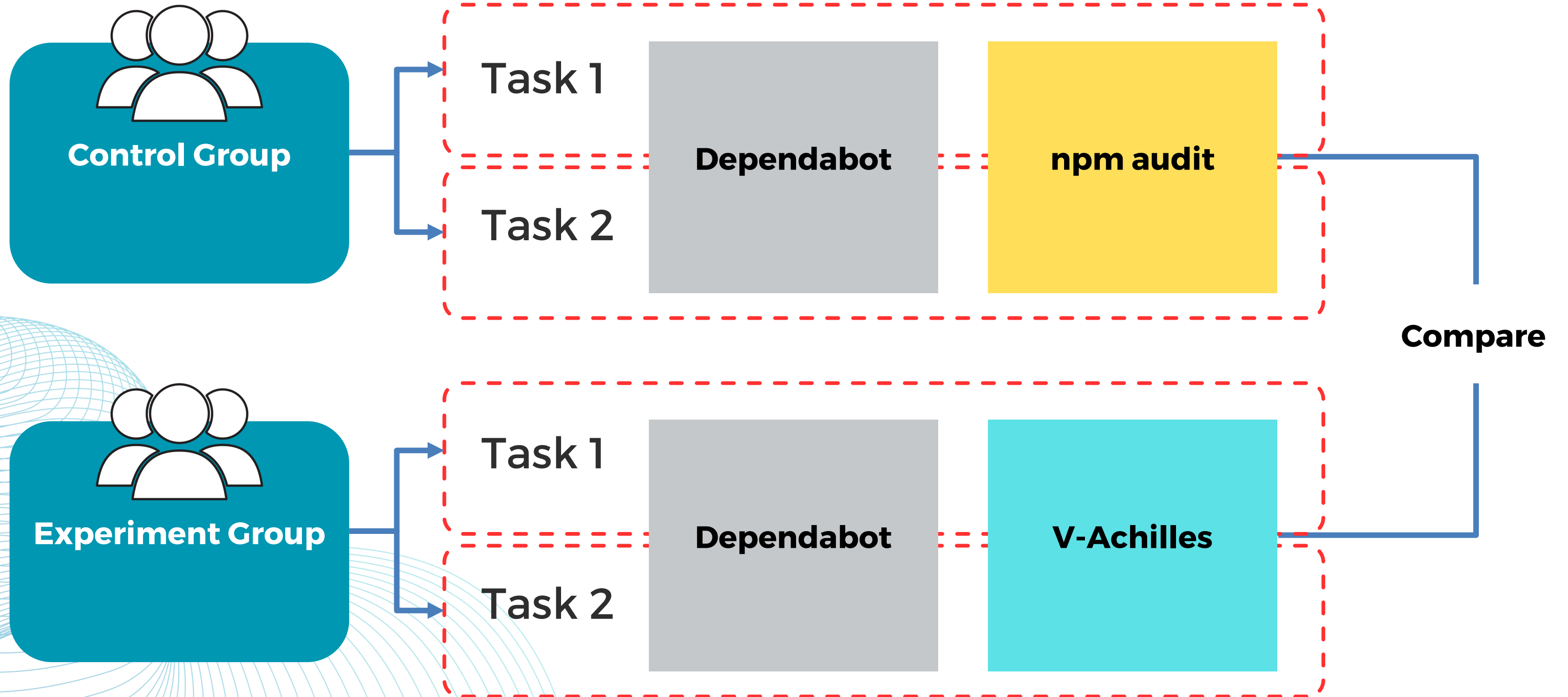
No	Dependency	Version	Severity	Type
1	netmask	2.0.0	High	Direct
2	base64-url	1.2.1	High	Transitive
3	angular-expressions	1.1.1	Low	Direct
4	minimist	1.2.0	Low	Transitive



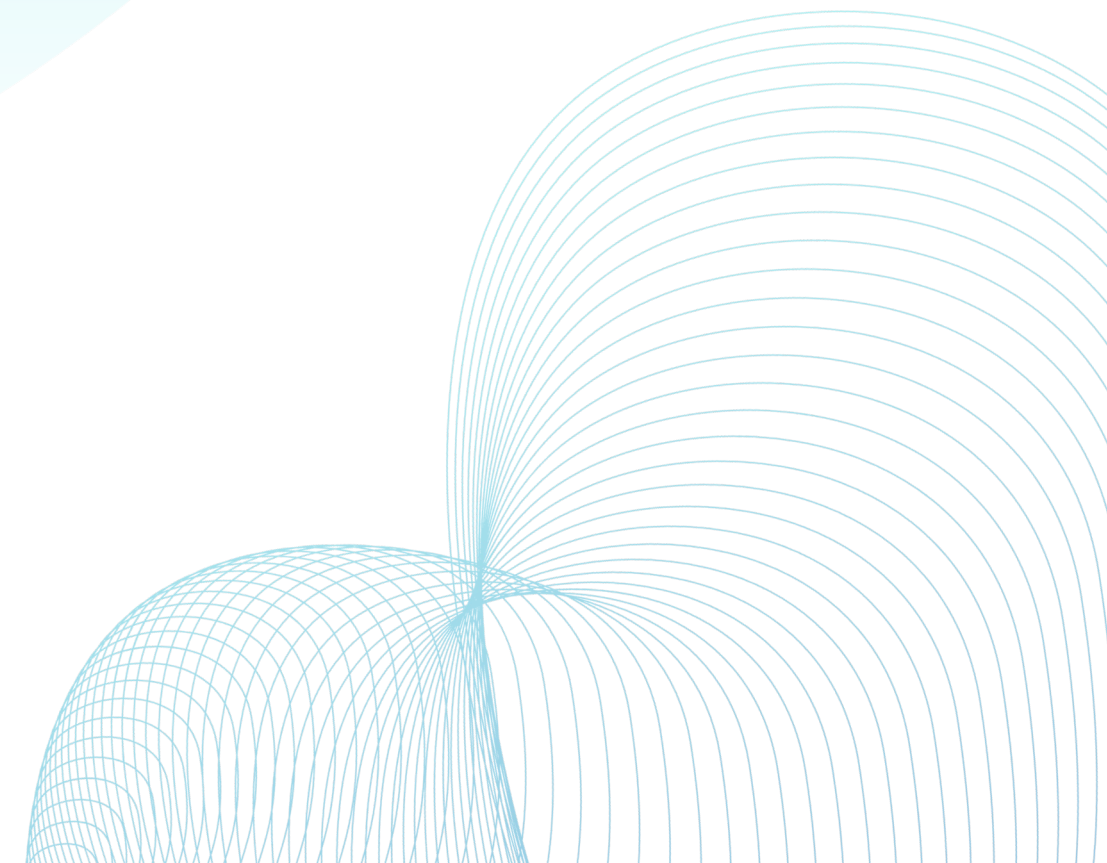
PARTICIPANTS' DEMOGRAPHIC AND TOOLS ASSIGNMENT

Group	Participants	Know Trans. Dep.	Tools Assignment
V-Achilles (Experimental Group)	E1	No	Dependabot followed by V-Achilles
	E2	Yes	
	E3	Yes	
	E4	No	
	E5	Yes	
	E6	No	
	E7	No	
	E8	No	
	E9	No	
	E10	No	
npm-audit (Control Group)	C1	Yes	Dependabot followed by npm audit
	C2	No	
	C3	Yes	
	C4	No	
	C5	No	
	C6	Yes	
	C7	No	
	C8	No	
	C9	No	
	C10	No	

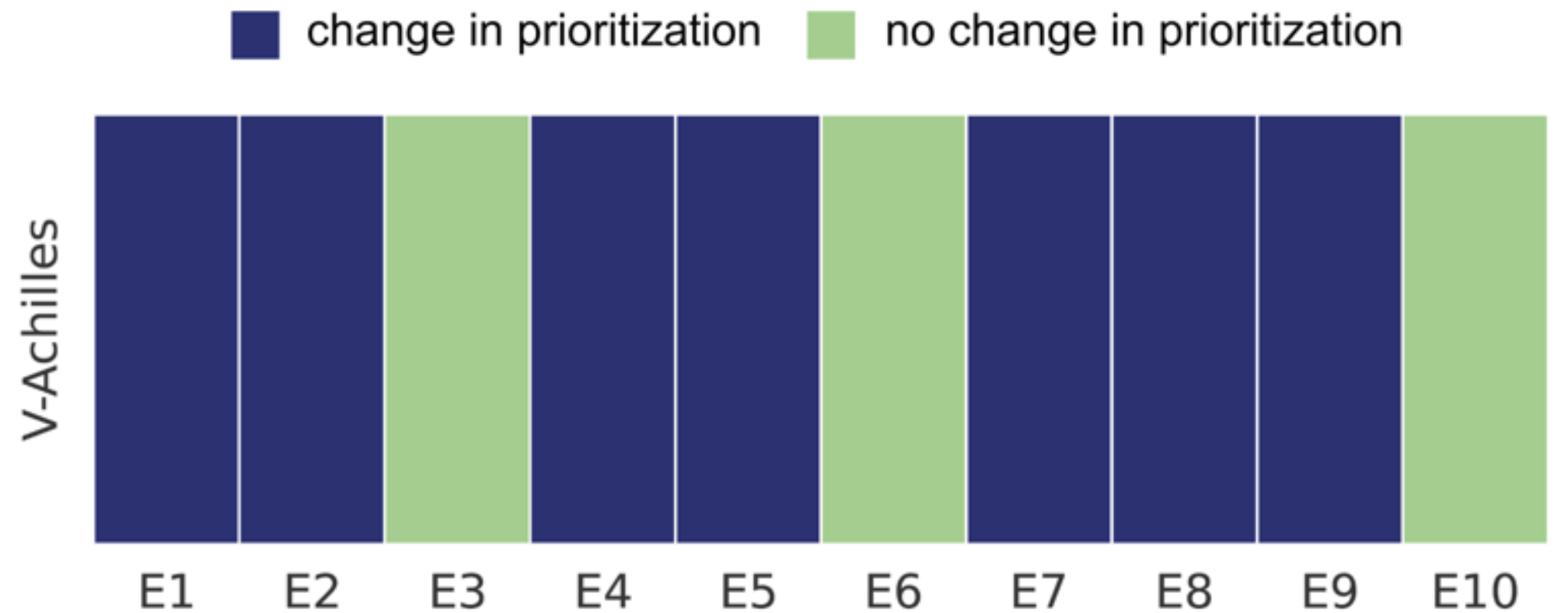
EXPERIMENTAL SETTINGS



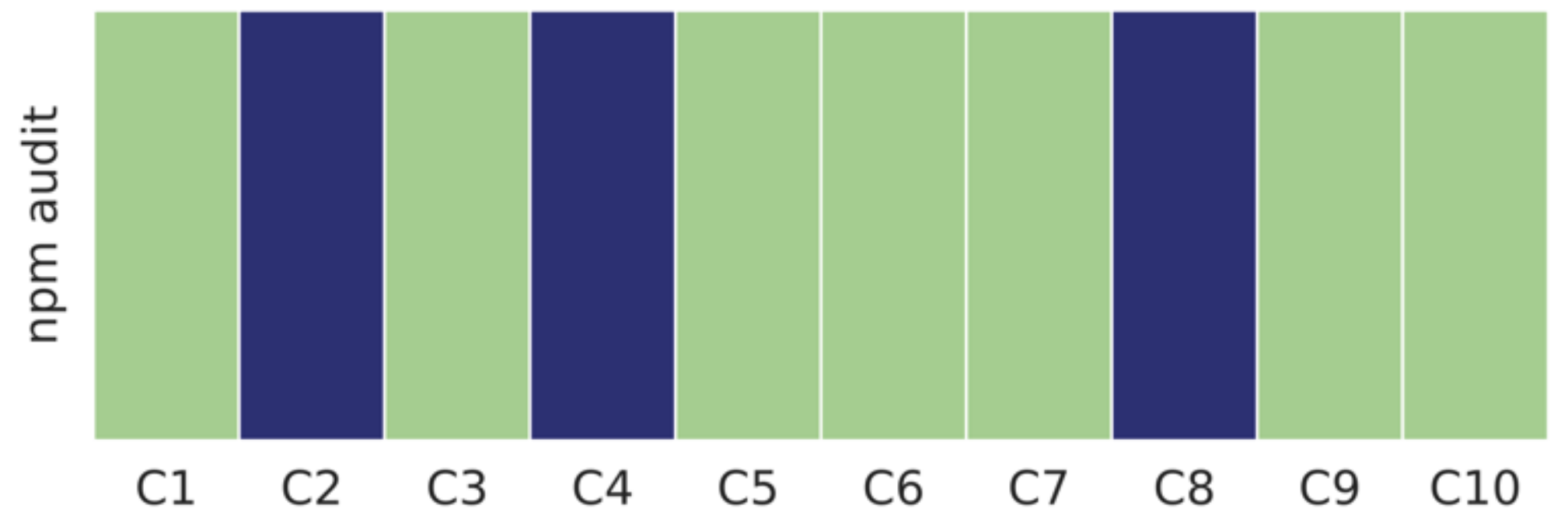
RESULTS



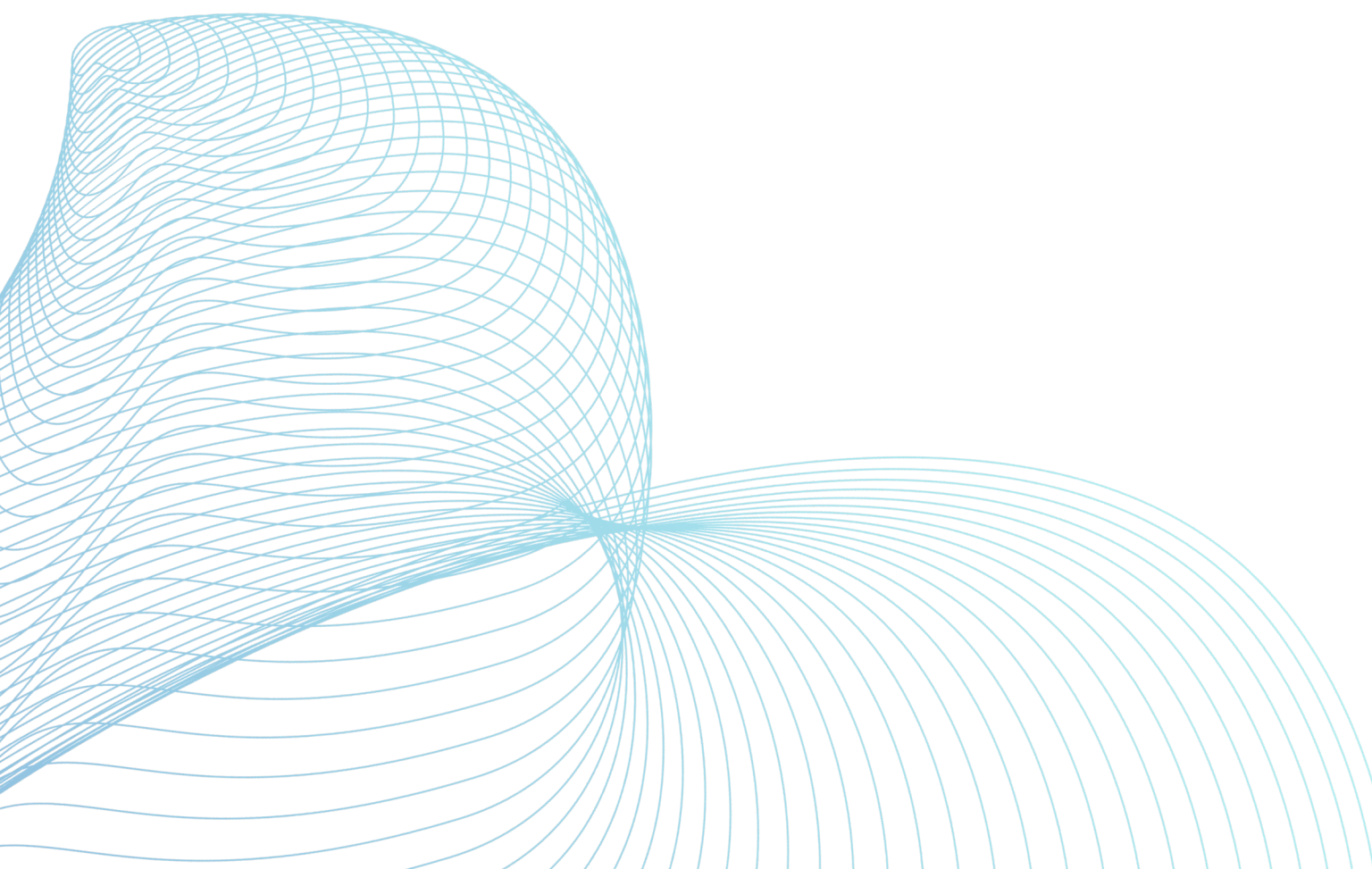
TASK 1: NAVIGATING DEPENDENCIES WITH COMPLEX GRAPHS



(a) Experimental Group: V-Achilles



(b) Control Group: npm audit

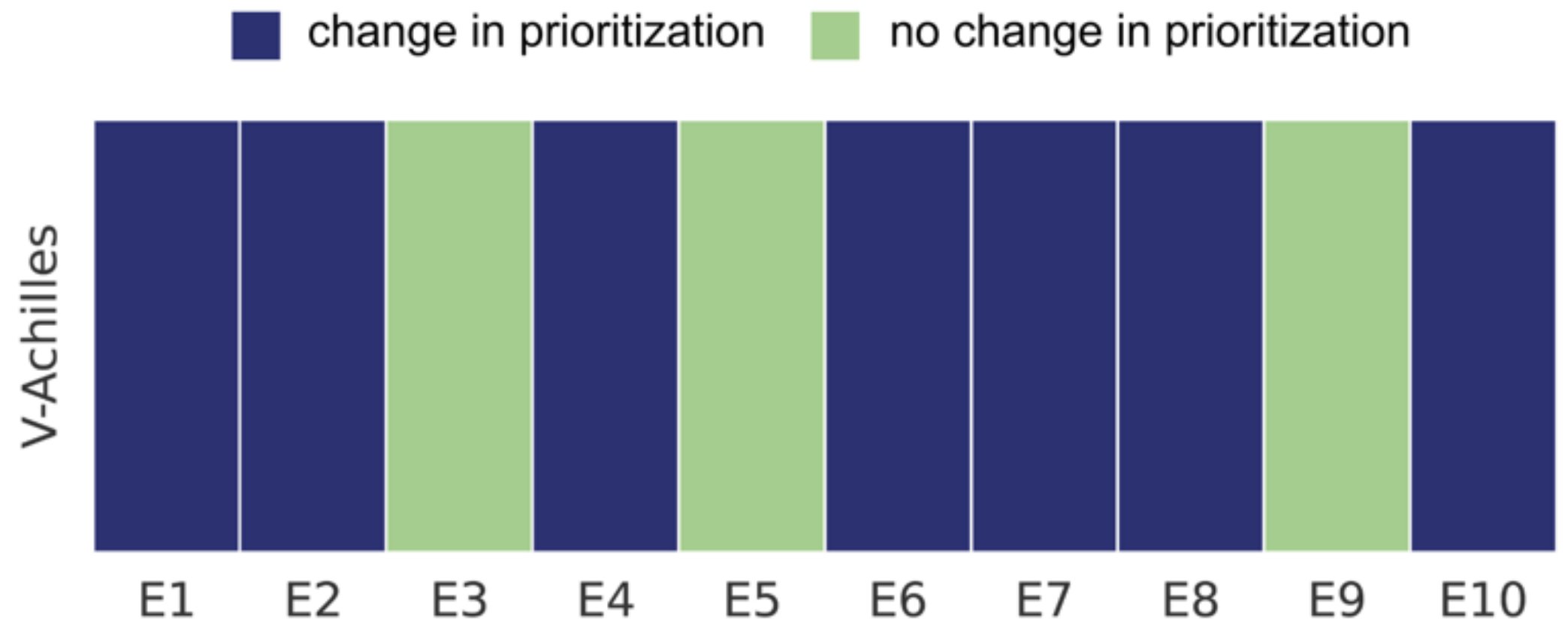
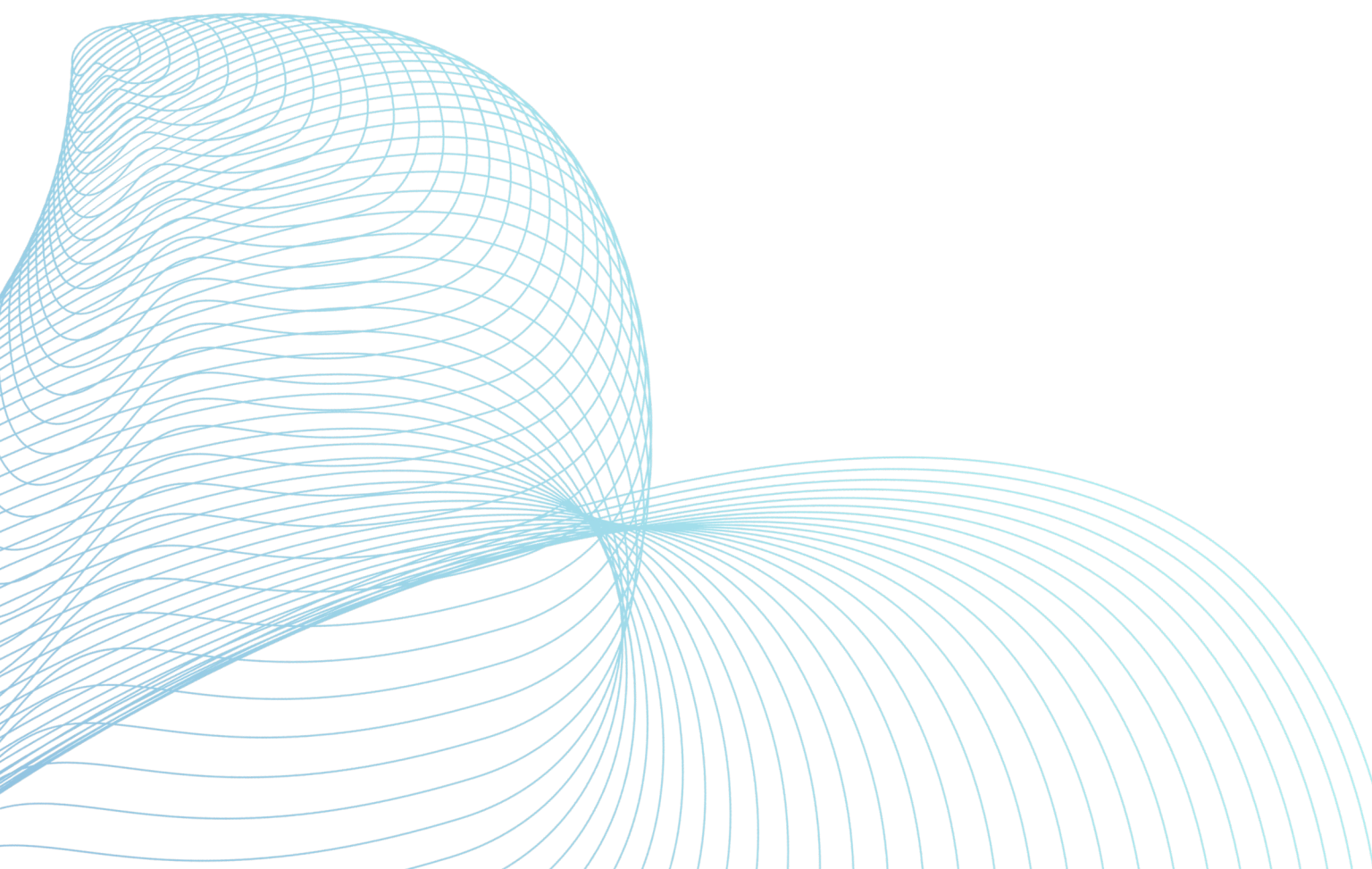


PARTICIPANTS FEEDBACK

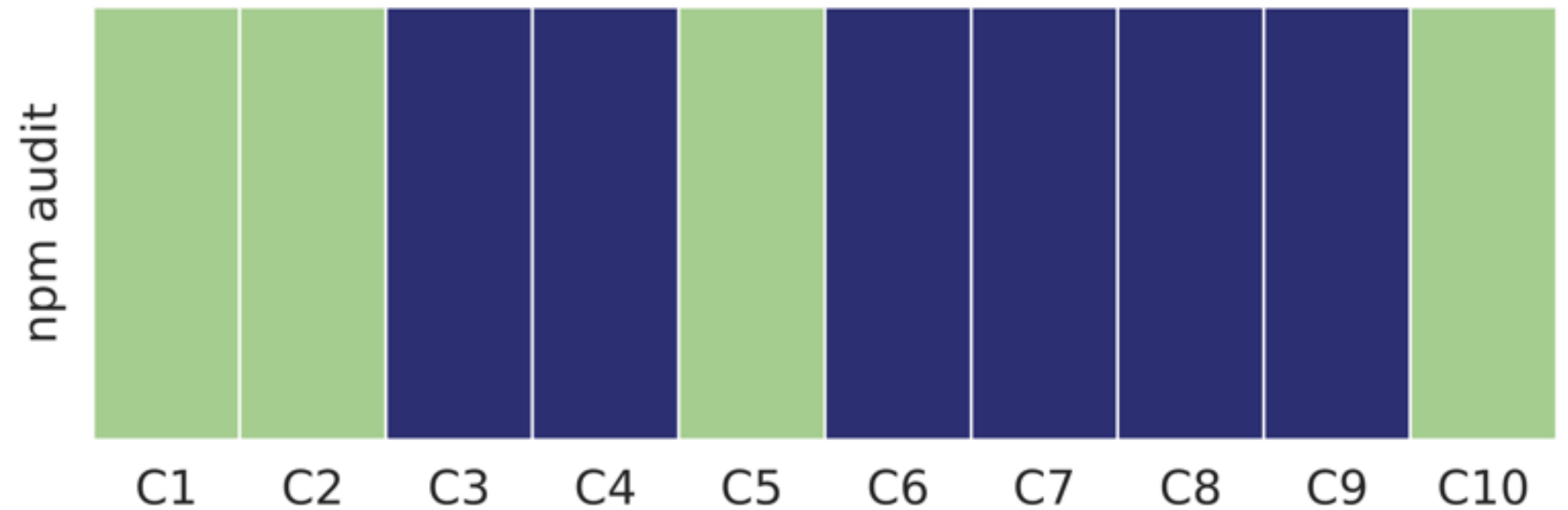
“I added more emphasis on high severity and **complex dependency** because of its complexity”.

“[After seeing V-Achilles’s visualization, I can see the] **number of transitive dependencies in each library**. If the number is high, it may interrupt other libraries once updated.”

TASK 2: NAVIGATING TRANSITIVE DEPENDENCIES WITH VULNERABILITIES



(a) Experimental Group: V-Achilles



(b) Control Group: npm audit

PARTICIPANTS FEEDBACK

“[After seeing the visualization] I checked **their severity and the dependency whether direct or not.** netmask and base64-url are high severity but netmask is direct dependency. **I think direct dependency is easier to fix than transitive dependency,** then I think it is the highest priority than others.”

SEE TO BELIEVE: USING VISUALIZATION TO MOTIVATE UPDATING THIRD-PARTY DEPENDENCIES

We study the effectiveness of a **dependency graph visualization (DGV)** to motivate developers to update vulnerable dependencies.

7 out of the 10 participants who used our visualization changed their prioritization in the two tasks of a project with vulnerable complex dependencies and a project with vulnerable direct and indirect dependencies.