Researcher: Christophe Cosse
Supervisors: Sebastian Proksch, Mehdi Keshan
Email: c.p.h.cosse@student.tudelft.nl
Date: 29th June 2021

Analysing the effectiveness of fine-grained dependency analysis to convince developers of updating their dependencies

TU Delft Delft University of Technology

1 Dependency Maintenance

- **Dependency Maintenance** is critical to assure **security** of large software projects
- New **vulnerabilities** are discovered every day
- Keeping up to date with these vulnerabilities manually is nigh impossible
- Automated software (e.g. **Dependabot**) do this automatically
- However these bots only analyse at **package-level** which results in lots of false positives

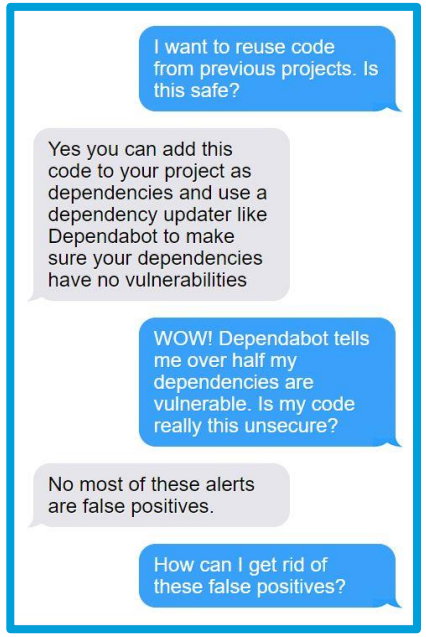
2 Fine-Grained Call Graphs

- The **FASTEN** project has a library to generate **calls graphs** at the **method level**
- Permits us to trace the calls in projects and see if the actual **vulnerable** methods are called
- FASTEN **Metadata Database** contains large collection of projects, vulnerable dependencies with the vulnerable methods.

3 Research Question

- Recent studies have shown fine-grained analysis to be more **accurate**
- But there hasn't been real research into the benefits for **dependency management**

> "Do people react to fine-grained information more than package-level information (Dependabot)?"



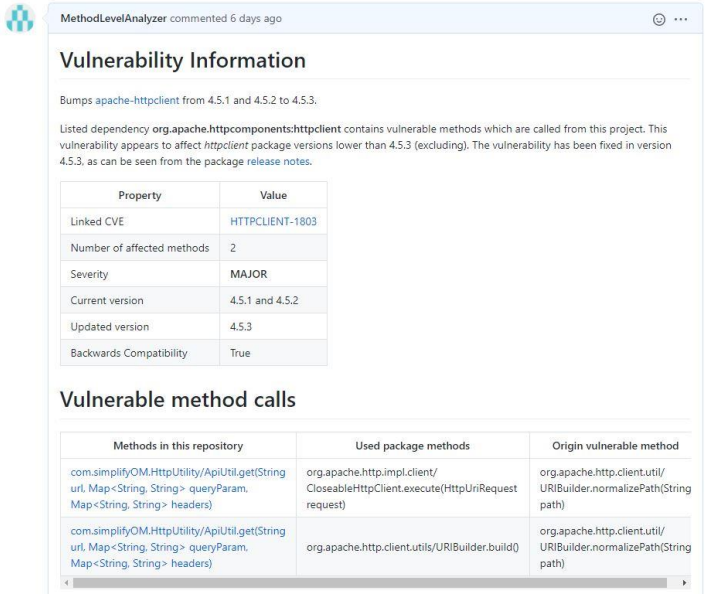


Figure 1: Extra method level information in security pull request

4 Methods of Research

- Identify vulnerable projects using the FASTEN project **library** and **Database**
- Generate **fine grained information** for these projects using call graph generation functionality of the FASTEN project
- Notify these projects with **issues** and **pull requests** containing this **fine grained information** and a survey to understand their experiences with the information
- Record the responses and analyse it to compare to understand whether this information helps.
- Use available **literature** to put into context whether this information really helps to convince developers to update their dependencies.

Table 1 shows the interactions from developers with the pull requests containing the method level information

Table 2 shows the responses to the survey in the pull request

Activity	Active	Inactive
# Pull Requests	25	12
# Merges	3	0
# Interactions	7	0
# Ticked Box	1	0
Average Time to Respond	10 days	NA
# Responses within 1 day	4	0
# Responses after 3+ weeks	3	0

Table 1: Obtained responses from PRs

5 Conclusions

- Not enough developers responded to the pull requests to be able to drawn any hard conclusions
- However data obtained is in line with the literature in terms of developers being unaware of vulnerabilities in their projects.
- The developers that did reply however did suggest that this extra fine-grained information helped with convincing them that the vulnerability did indeed affect their project
- However the extra information doesn't seem to make dealing with the vulnerability easier
- Nevertheless 2 of the developers did indicate they would like to receive more security pull requests with extra fine-grained information
- Further data collecting is therefore required to be able to certify these results and answer the research question.

Question	Yes	No
I was aware of the vulnerability affecting my project before being informed by the Pull Request.	0	3
I was convinced by the provided method call data that the vulnerability indeed affects my project.	2	1
I plan on merging the PR in the near future.	3	0
The provided method call information has made my process of dealing with the vulnerable dependency easier	1	2
I have given priority to the task of fixing the vulnerability over other project tasks that are yet to be completed.	2	1
I would like to receive this kind of method information in future vulnerable dependency Pull Request descriptions.	2	1

Table 2: Obtained responses from Survey