

FASTEN: Fine-grained Analyses of Software Ecosystems as Networks

Sebastian Proksch, Delft University of Technology

The leftpad incident

- A developer removed an NPM library, consisting of just 11 lines of code, over a naming dispute.
- The web broke in response.

The Register®

{* SOFTWARE *}

How one developer just broke Node, Babel thousands of projects in 11 lines of JavaScript

Code pulled from NPM – which everyone was using

Chris Williams, Editor in Chief Wed 23 Mar 2016 // 01:24 UTC

UPDATED Programmers were left staring at broken builds and failed installations on Tuesday after someone toppled the Jenga tower of JavaScript.

A couple of hours ago, Azer Koçulu unpublished more than 250 of his modules from [NPM](#), which is a popular package manager used by JavaScript projects to install dependencies.

Koçulu yanked his source code because, we're told, one of the modules was called [Kik](#) and that apparently attracted the attention of lawyers representing the instant-messaging [app](#) of the same name.

According to Koçulu, Kik's briefs told him to rename the module, he refused, so the lawyers went to NPM's admins claiming brand infringement. When NPM took Kik away from the developer, he was furious and unpublished *all* of his NPM-managed modules. "This situation made me realize that NPM is someone's private land where corporations are more powerful than the people, and I do open source because Power is with The People," Koçulu blogged.

Unfortunately, one of those dependencies was [left-pad](#). The code below. It pads out the lefthand-side of strings with zeroes or spaces. thousands of projects including Node and Babel relied on it.

With [left-pad](#) removed from NPM, these applications and widely used pieces of open-source infrastructure were unable to obtain the dependency, thus fell over during development and deployment. Thousands, worldwide. [Left-pad](#) was fetched 2,486,696 times in just the last month according to NPM. It was that popular.

The equifax disaster

- Security breach through vulnerability in Apache Struts dependency
- Details stolen for 143M user accounts
- Estimates of >\$4B damages
- Patch was available for more than two month



LILY HAY NEWMAN

SECURITY 09.14.2017 01:27 PM

Equifax Officially Has No Excuse

A patch that would have prevented the devastating Equifax breach been available for months.

LA TIGRE FOR WIRED

CAPPING A WEEK of incompetence, failures, and general shady behavior responding to its massive data breach, Equifax has confirmed that attack system in mid-May through a web-application vulnerability that had a patch in March. In other words, the credit-reporting giant had more than two months to take basic security precautions that would have defended the personal data of 143 million people from being exposed. It didn't.

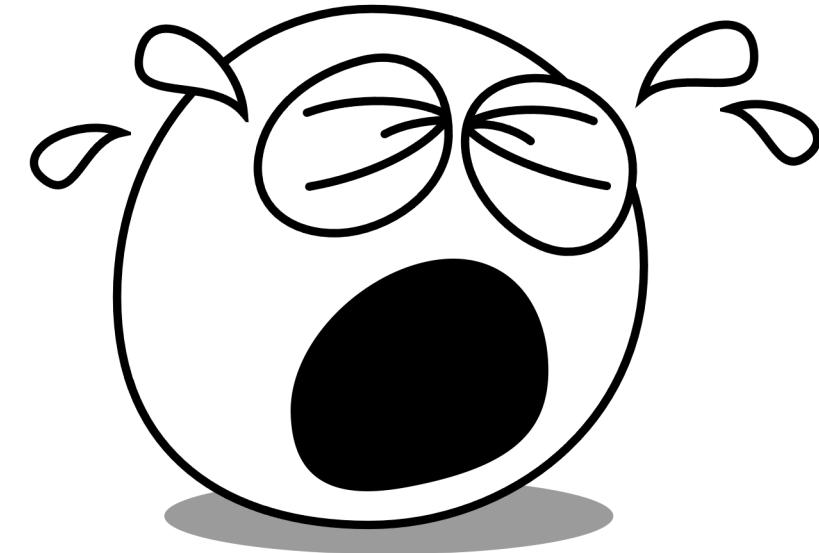
As the security community processes the news and scrutinizes Equifax's posture, numerous doubts have surfaced about the organization's commitment to data steward. The company took six weeks to notify the public after first learning of the breach. Even then, the site that Equifax set up in response to offer free credit monitoring was itself riddled with vulnerabilities. And when journalist Brian Krebs first reported, a web portal for handling credit-reporting requests from customers in Argentina used the embarrassingly inadequate credentials "admin/admin." Equifax took the platform down on Tuesday. But ongoing discoveries increasingly paint a picture of negligence—especially the failure to protect itself against a known flaw with a ready fix.

Untamed Use of Dependencies



The Sorry State of the Art

- Not much beyond simple package version matches
- No support for assessing updates
- No support for making decisions on which libraries to use
- No support for maintainers



We need to do better than that!

H2020 EU Project: FASTEN



Delft University of Technology



Athens University of
Economics and Business



University of Milano



Software Improvement Group



XWiki



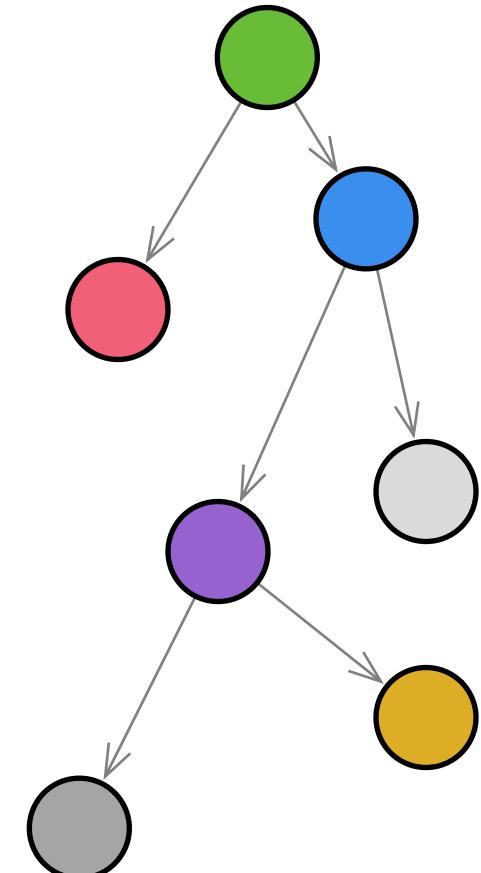
Endocode



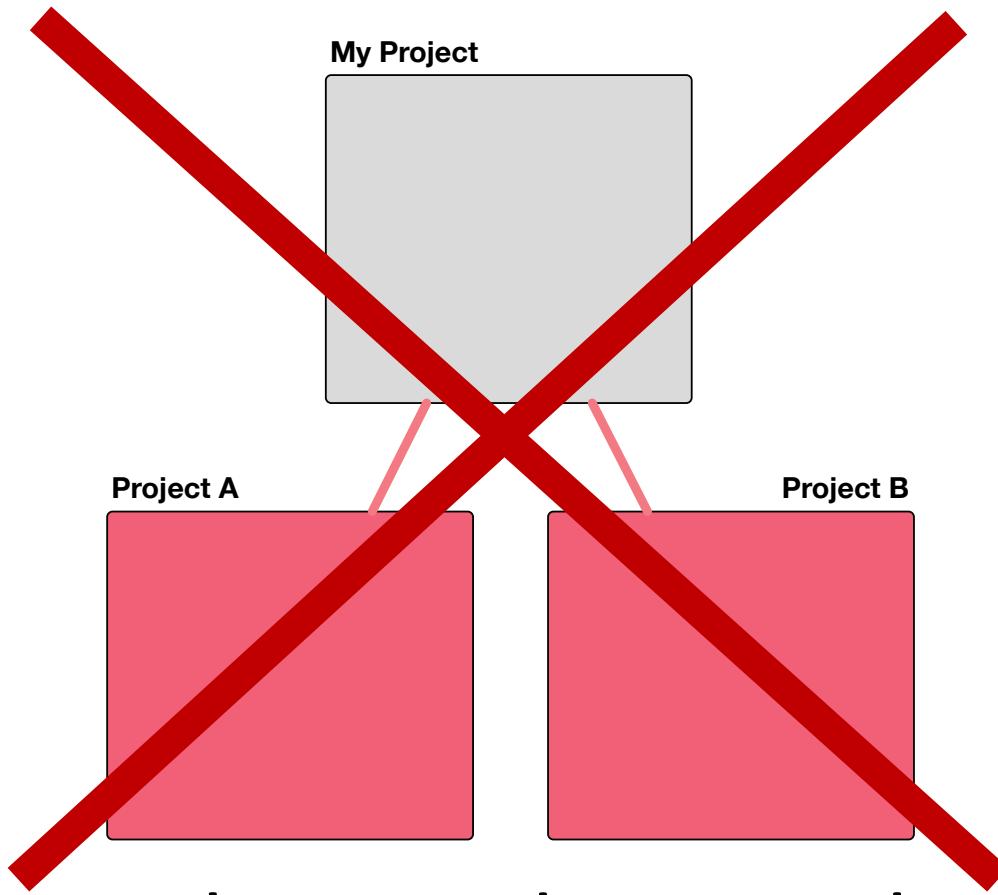
OW2

FASTEN: Revolutionize Dep. Management

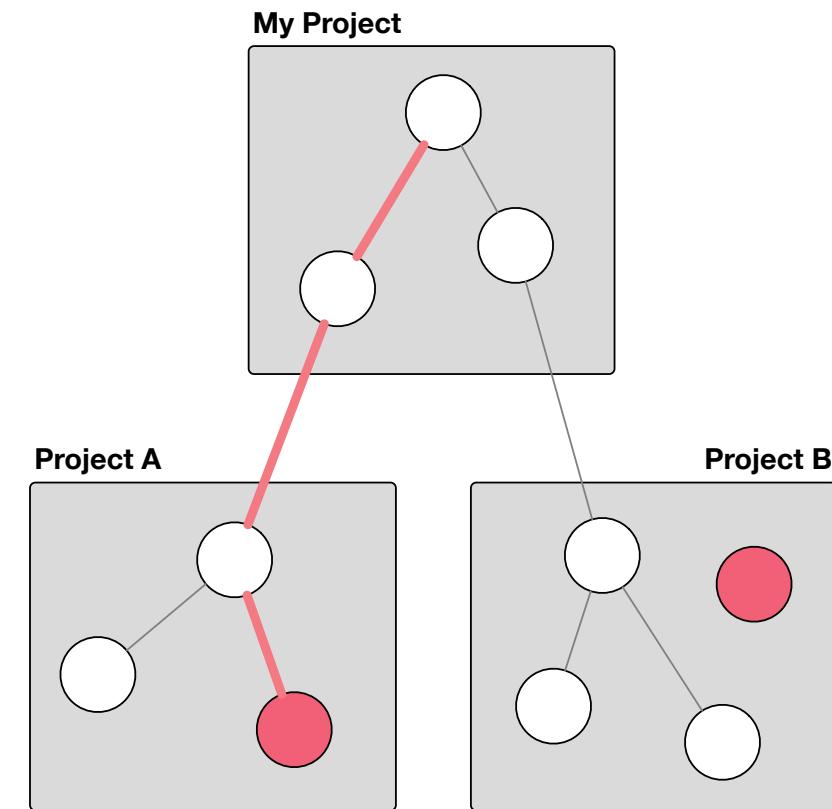
- Improve Vulnerability Detection
- Reliable Impact Analysis
- Better License Management



Dependency Networks



Package Dependency Network



Call Dependency Network

Our Research Goal: Static Analyses at Scale

- Scale Call-Graph Generation
- Use Cases
 - Licensing
 - Impact analysis
 - Vulnerability Detection
- Infrastructure



Incremental Call-Graph Generation

1. Resolve

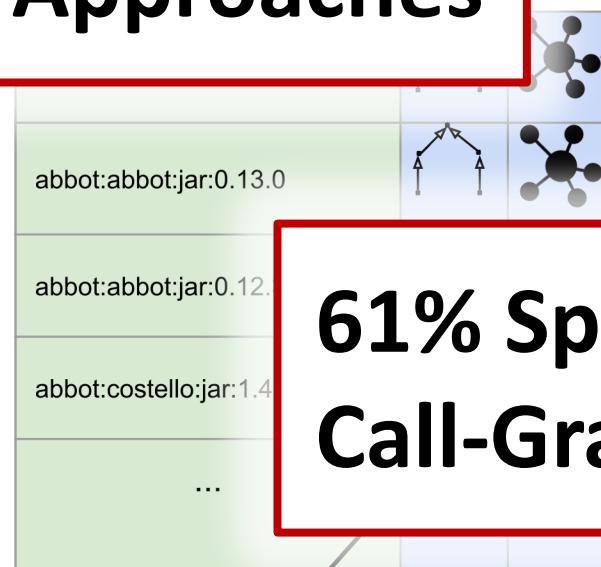
2. Compute (and cache)

3. Merge Into
Complete CG

**Comparative Accuracy to
State of the Art Approaches**

Dependency Set

- HTTPClient-0.3-3.jar
- abbot-0.13.0.jar
- Abbot-0.12.3.jar
- costello-1.4.0.jar



**61% Speed-up on Cached
Call-Graph Generation**

Type Hierarchy

Call Sites

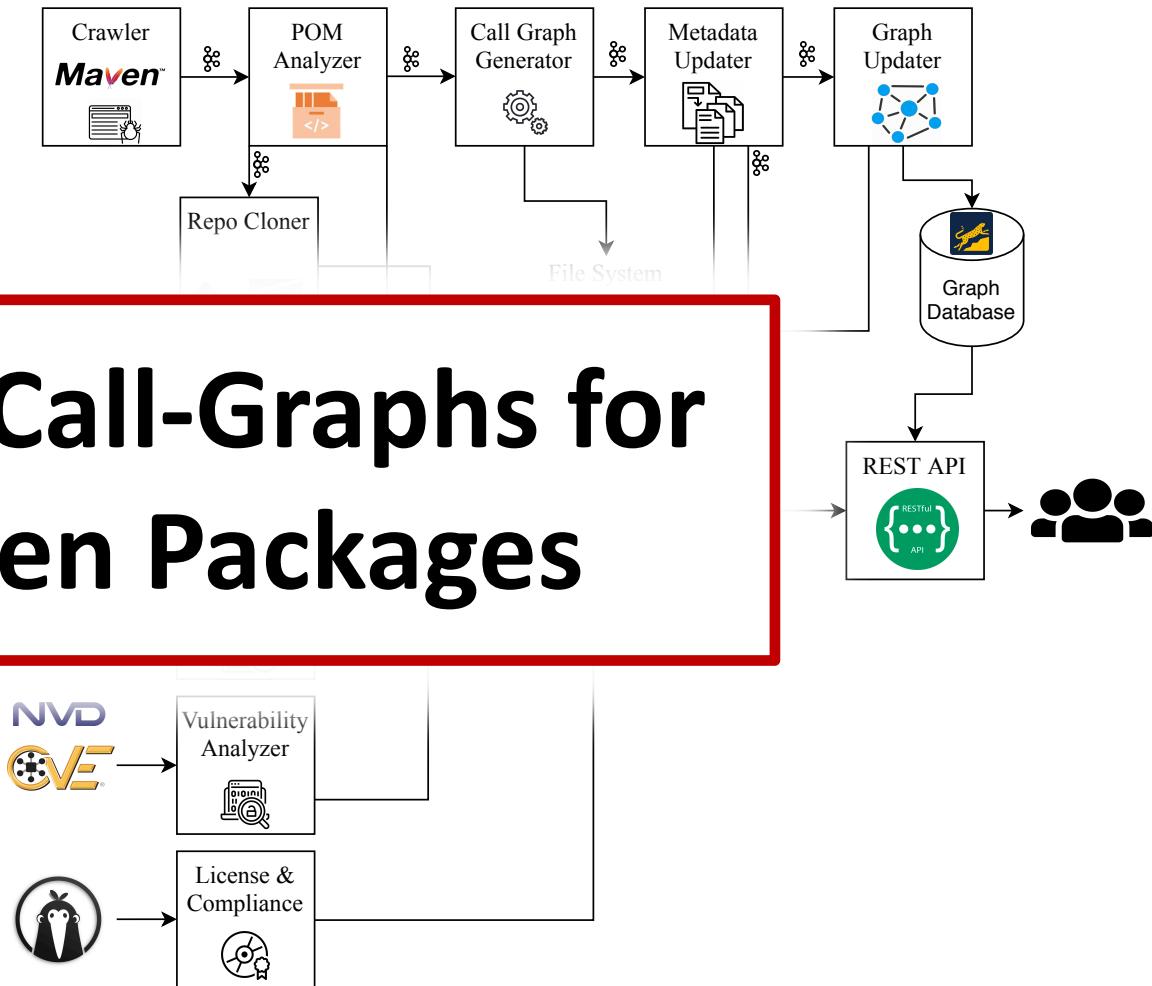
Data Processing Pipeline

Kubernetes Cluster

- 4 Machines
- 120 Cores
- 1.25 TB
- 233 TB

We have Call-Graphs for
3M+ Maven Packages

Apache Kafka



Browsable Dependency Information

The screenshot displays two views of the FASTEN application. The left view shows the main homepage with a search bar and a grid of package cards. The right view shows a detailed page for the JUnit 4.12 package, listing vulnerabilities and their dependencies.

FASTEN proof-of-concept

Learn about robustness and security of the packages in your software ecosystem.

Search package

JUnit
The programmer-friendly testing framework for Java and the JVM.

JBossMQ Client
Implementation of the Java Message Service API part of the J2EE specification.

Mockito Core
Tasty mocking framework for unit tests in Java.

SLF4J API
The Simple Logging Facade for Java.

Scala Library
Standard library for the Scala Programming Language.

Learn more about [concept](#) and [use cases](#).

Disclaimer: FASTEN is currently under development with a limited amount of packages available.

JUnit 4.12 View

Modules Vulnerabilities Versions

Vulnerabilities

AcknowledgmentRequest.init()
→ JavaLang.Parse()
→ ParserJackson.ParseJson()

Connection.newThread()
→ EasyThreading.threadRequest()

DurableSubscriptionID.getClientId()
→ UserRequest.getUser()
→ NumberGenerator.generateID()

Build-System Integration



Stay Tuned & Get Ready for Public Testing



@FastenProject



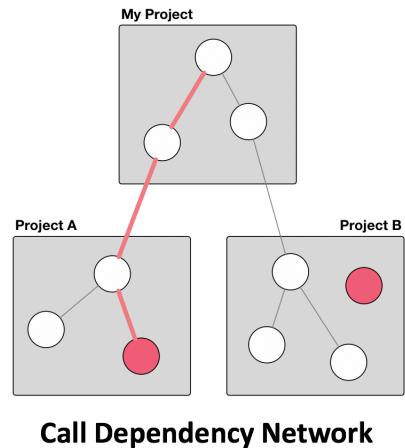
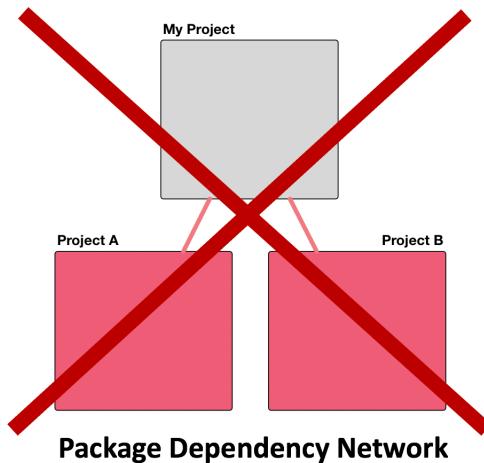
<https://github.com/fasten-project>



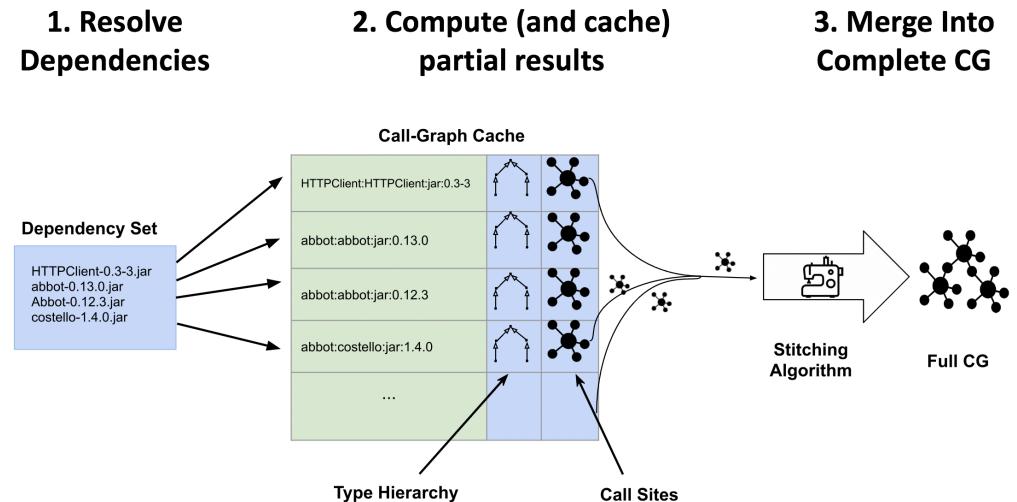
<https://www.fasten-project.eu/>

Summary

Dependency Networks



Incremental Call-Graph Generation

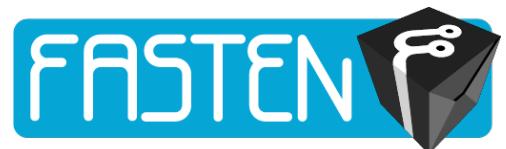


Sebastian Proksch

S.Proksch@tudelft.nl

Delft University of Technology

www.fasten-project.eu



Build-System Integration

