

Veracity Technology Spearhead

Enabling end-to-end veracity within value exchange ecosystems

Improving Software Supply Chain Security: Discovering Blind Spots in Software Composition Analysis

Jens Dietrich, Tim White – Victoria University Of Wellington, Shawn Rasheed – Te Pūkenga, Alex Jordan – Oracle Labs Austria



The Problem

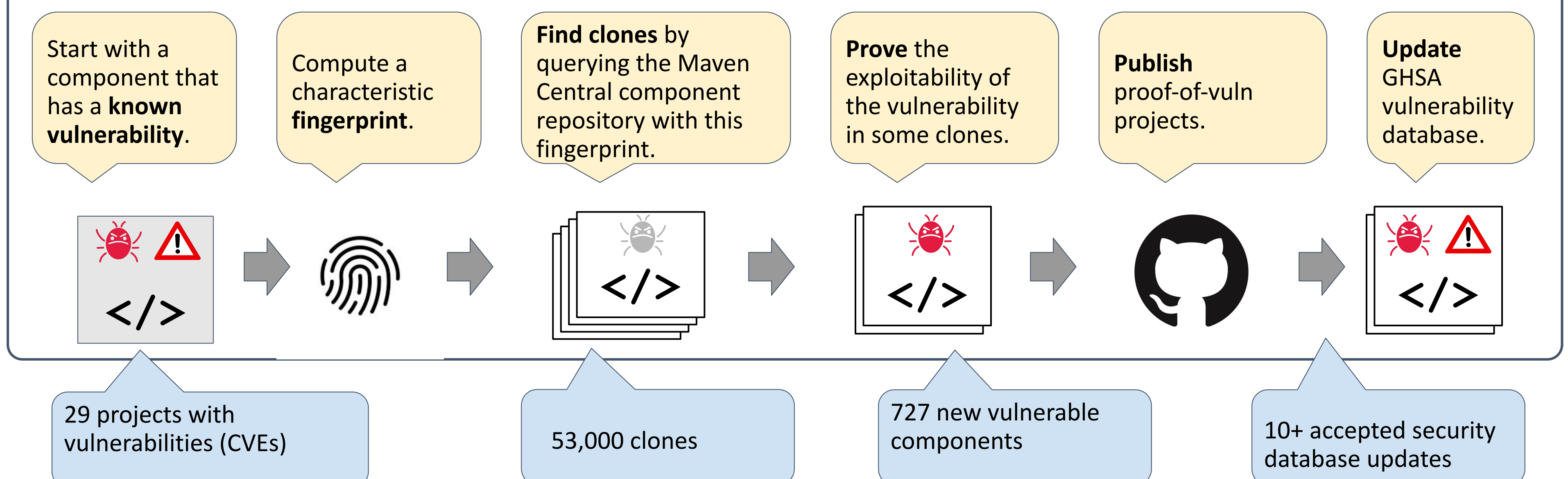
Modern software is built from components to achieve economy of scale. Vulnerabilities in those components can affect applications and be exploited.

This has led to cyber attacks causing significant economic and social harm. Famous incidents include *log4shell*, *heartbleed* and *equifax*.

In response, *software composition analysis* tools have been developed to detect such vulnerabilities, and those tools are now widely used.

However, software composition analysis tools have *blind spots*: they miss vulnerabilities.

Our Approach: An Automated Pipeline From Analysis to Impact



Evaluation

We have analysed 29 components with vulnerabilities (including log4shell). We have retrieved over 53k potential vulnerable clones from Maven, and after running our analysis on this set, we detected 727 vulnerable clones. We have demonstrated that industrial and open source software composition analysis tools fail to detect most of those.

Impact

For each vulnerability found, our tools generated and released a proof-of-vulnerability project, published here: <https://github.com/jensdietrich/xshady-release/>

We have updated one of the main vulnerability databases (The GitHub Security Advisory) with our results via pull requests. So far, 10 pull requests (each for multiple components) have been accepted.

Example: <https://github.com/github/advisory-database/pull/2258>

As existing software composition analysis tools use this database, our results have immediately improved industrial practice.

For more information please email jens.dietrich@vuw.ac.nz