# AboutCode
# VulnerableCode/Vulntotal: Browser Extension (Category B)
# Google Summer of Code 2024 Project Proposal

Malik Akbar Hashemi Rafsanjani
malikakbarrafsan@gmail.com
https://site.malikrafsan.tech/
Bandung Institute of Technology
Bandung, Indonesia
+62-853-2650-2042

## Abstract

AboutCode is an open-source community providing tools for Software Composition Analysis (SCA). One of the tools provided by AboutCode is Vulntotal, which cross-validates the vulnerability coverage of publicly available vulnerability check tools and databases. This project will focus on developing a browser extension running Vulntotal on the browser and query the vulnerability data sources for comparing them.

## Project Information

- Project Size: Medium (175 hours)
- Link to The Original Project Idea: https://github.com/nexB/aboutcode/wiki/GSOC-2024-Project-Ideas/#vulnerablecodevulntotal-browser-extension-category-b
- Related Issues: https://github.com/nexB/vulnerablecode/issues/1121

## Project Description

VulnerableCode is AboutCode's project that provides an open database of software packages that have security vulnerabilities (vulnerable packages). This project has multiple functionalities, such as collecting, aggregating, and correlating known security vulnerabilities with a correct package version. However, there are instances where a package is reported as having security vulnerabilities by some tools but not by others. The VulnerableCode project also provides a subproject, VulnTotal, to cross-validate public vulnerability check tools and data sources. Currently, VulnTotal provides CLI tools that take a PURL argument and return vulnerability data from various data sources.

This project aims to expand the usage of VulnTotal by providing a browser extension that runs VulnTotal. This project will allow a browser user to query vulnerability data sources and compare them directly on the client side of the browser. The same as VulnTotal, this extension will get a

PURL input from the user and provide the vulnerability data. By developing this project, a better user experience can be provided to cross-validate and check vulnerability data sources.

This project is done by implementing a browser extension that can run Python code. Python is chosen as the VulnTotal project is implemented using Python. By using Python directly in the browser extension, the existing project doesn't need to be rewritten. Consequently, this project requires us to research a suitable tool to run Python code in the browser and run the VulnTotal on that tool.

## Key Deliverables

This project has several key deliverables as follows.

- Research results of the suitable tools to run Python code in the browser, especially for a browser extension
- A browser extension that is capable of running the VulnTotal on the client side and querying the vulnerability data sources to compare them

## Innovation and Contribution

This project aims to improve web browsing security by providing users with a convenient and accessible way to assess website vulnerabilities. Traditional vulnerability assessment often involves direct terminal usage. This extension provides better UX by integrating vulnerability scanning directly within the browser. Users can assess website security on the fly, without switching between tools or contexts.

In essence, this browser extension contributes to the VulnerableCode project by expanding its reach to a wider audience by eliminating technical barriers. It improves user experience by providing a seamless and convenient way to check website security directly within the browsing workflow. Finally, it promotes security awareness by simplifying vulnerability data and raising user consciousness about potential online threats.

## Project Implementation

### Python Runner on Browser

Currently, many tools provide functionality to run Python code directly on the browser, such as

- PyScript
- Pyodide
- PyOxidizer
- Brython
- Pyjs

However, there are a few precedence projects of browser extensions using those tools to run Python code. Some of those tools are already assessed to be used on browser extensions. One successful trial is using PyScript and Pyodide as the runtime. Here is the URL to a minimum viable product (MVP) as proof of concept that Python code can be run on the browser extension. This browser extension enables the user to calculate simple addition and subtraction using Python code also providing Python REPL.

- https://github.com/malikrafsan/python-web-extension

### Running The VulnTotal

Some Python runner tools, especially Pyodide, already provide the functionality to import Python packages and custom Python files. This is very useful as the VulnTotal is already implemented and can be directly used on this project. Some of the references to the import functionality can be found here.

- Loading custom Python code: https://pyodide.org/en/stable/usage/loading-custom-python-code.html
- Loading packages: https://pyodide.org/en/stable/usage/loading-packages.html

## Proposed Schedule

#### March 18th - April 2nd (Application Period)

Writing the project proposal, asking for reviews from mentors, and trying to improve the content based on their feedback.

#### April 2nd - May 1st (Small Contributions and Initial Community Bonding)

- Get to know more with the AboutCode community
- Help to fix good first issues
- Attend routine AboutCode's meetings

#### May 1st - May 26th (Community Bonding Period and Cross-Validate Tools)

- Official GSoC period to get to know more with the AboutCode community, especially the mentor
- Research other tools that offer similar functionalities, especially in the context of browser extension
- Implement minimum viable product (MVP) for suitable tools
- Compare and analyze the pros and cons of each of the suitable tools and report it to the mentor
- Get to be more familiar with the VulnTotal project by examining directly the codes

#### May 26th - June 9th (Preliminary Package Porting)

- Importing small packages/custom Python code to the browser extension
- Analyze the importing capabilities of the tools and report to the mentor

### June 9th - July 5th (Importing VulnTotal into Extension)

- Import the VulnTotal codes to the browser extension
- Analyze the capabilities/setback because of the importing

### July 5th - July 12th (Midterm Evaluation)

- Prepare midterm evaluation documents.
- Request for mentor’s feedback and improve the content based on the feedback.

### July 12th - July 19th (Regress VulnTotal on Browser)

- Regress test of the VulnTotal codes on the browser extension
- Make sure all the VulnTotal’s capabilities run smoothly

### July 19th - August 19th (Improving UI/UX and Documentation)

- Develop better UI/UX for the user of the browser extension
- Make sure the UI/UX is intuitive and easy to use
- Documenting the project

### August 19th - August 26th (Final Week)

- Prepare all required GSoC documents.
- Request for mentor’s feedback and improve the content based on the feedback.
- Submit all necessary GSoC documents

## References

- VulnerableCode documentation: https://vulnerablecode.readthedocs.io/
- VulnTotal documentation: https://rtd.keshav.space/en/latest/
- PyScript documentation: https://pyscript.net/
- Pyodide documentation: https://pyodide.org/en/stable/
- Proof of concept of browser extension: https://github.com/malikrafsan/python-web-extension
- Original Project Idea: https://github.com/nexB/aboutcode/wiki/GSOC-2024-Project-Ideas/#vulnerablecodevulntotal-browser-extension-category-b

# Personal

## Personal Information

- **Name**: Malik Akbar Hashemi Rafsanjani
- **Country / Timezone**: Indonesia / UTC+7
- **Email**: malikakbarrafsan@gmail.com
- **Gitter Username**: @malikrafsan:matrix.org
- **GitHub Profile**: https://github.com/malikrafsan

- **Resume**: https://bit.ly/ResumeMalikAkbar
- **PPT Professional Profile**: https://bit.ly/ProfessionalProfileMalikAkbar

## Motivation

My name is Malik Akbar, and I am a final-semester Computer Science student at Bandung, Institute of Technology, Indonesia. I have a very big passion for software engineering in general. I started learning in a high-level domain (web development, mobile development, and backend development). You can also find my full resume attached in the personal information section.

In my recent internship, I have interacted a lot with the security team of the company and I am very interested in contributing to security-related open-source programs. I am very intrigued to contribute to AboutCode, especially the browser extension project as I can hone and channel my web development skills and also my interest in security-related projects. I believe this would be a very great opportunity for me to learn more about security projects and learn to contribute to open source at the same time.

## Experiences

My professional experience encompasses various technical experiences from internships and competitions. At Grab-OVO, I was a software engineer intern, tasked with migrating essential services to Kubernetes. My internship at OCBC Bank involved developing a vital internal dashboard using React and Golang. Furthermore, at Payable (YC 2022), I contributed to refining payment and shipping integrations using Golang and GCP. Not only those companies, but I also worked for GDP Labs, Advotics, and eComindo.

During my internship at GDP Labs, I helped to develop an online customer service real-time chat app for the biggest private bank in Indonesia and an automation reporting app, integrated with a work management platform, Asana. At Advotics, I also developed microservices SaaS apps with high modularity for various specific client companies using Spring Boot. Also, I worked as a developer to develop a real-time collaboration platform web application with multiple game apps when I was an intern at eComindo using Next JS. These experiences have endowed me with a deep understanding of software engineering technologies, especially web-related.

Beyond my working experiences, I also gained various honors and awards on national and international stages. I achieved the Runner-Up position in Hackathon competitions at the Southeast Asia level twice from Garuda Hacks 2.0 and Garuda Hacks 4.0. I also secured the 2nd Runner-Up spot at the Schneider Go Green Regional Final in Southeast and East Asia level and earned 3rd Position at the Asia Pacific Level in the International RoboCup MSL Asia Pacific Scientific Challenge 2021. I also received the National Winner in a data science competition (Gemastik) and was awarded as the National Champion in Schneider Go Green.

## Skill Set

- **Programming Languages**: Python, Golang, TypeScript, Java, C#, C/C++, PHP
- **Front-end Tools**: React JS, Vue JS, Next JS, Svelte, Tailwind, Vanilla JS
- **Back-end Tools**: Django, FastAPI, Spring Boot, Express, MySQL, PostgreSQL, Docker, Kubernetes, Terraform,

## Acknowledgment

- Do you plan to have any other commitments during GSoC that may affect your work?
    - Currently, I am a final-year computer science student, but I don't have any courses left, except my final thesis. Besides my final thesis, I don't have any other commitments during GSoC periods.
- Any vacations/holidays?
    - I don't have any planned vacations/holidays during the GSoC periods.
- Will you be available full-time to work on your project?
    - Yes, I am available full-time to work on this project as I have planned for this summer period to be fully committed to the GSoC program.
- We also have weekly status meetings, same time as the community call, on Mondays, would you be able to attend them?
    - Sure, I would be able and more than happy to join the meetings
- We will be following the 12-week standard coding period as default for all our, projects, unless unforeseen circumstances arise. Do you accept the standard coding period as default?
    - Yes, I accept this period and I believe I can complete this project on time