Extract: A PHP Foot-Gun Case Study
Static Analysis Artifact

Jannik Hartung

26.05.2025

The artifact contains two contributions:

1. The source code of the static analysis on PHP Code-Property-Graphs
2. A list of PHP projects containing exploitable `extract` calls

The static analysis tool can be reused on new projects and is packaged in a Docker image. The tool can be run locally to verify the workflow of the automatic part of the analysis in the paper is functional. Future work can use the static analysis implementation and result analysis to derive their own analysis on PHP code.

The list of exploitable `extract` calls was created manually based on the results of the static analysis and cannot be reproduced as part of this artifact evaluation. Thus the focus of evaluating the functionality of the artifact is on the static analysis tool.

The artifact can be downloaded from https://doi.org/10.5281/zenodo.15526425.

1 Static Analysis Tool

The static analysis uses the PHP-CPG framework by *Wessels et al.*, which provides the tools to generate Code-Property-Graphs of a large number of PHP projects and run custom analysis passes on them. This artifact contains our analysis pass to find `extract` calls in a CPG and search for a data-flow connection between user input and the `extract` call.

The artifact review focused on running the static analysis and verifying its ability to identify vulnerable `extract` calls. A script to clone sample PHP projects from GitHub and run the evaluation on them is provided in the artifact.

Please refer to the `README.md` in the `extractor` directory for detailed instructions on how to build and run the tool.

2 Vulnerable PHP Applications

The list of exploitable `extract` calls, `Extract_Project_Exploitabililty_Results.csv`, was produced by manually inspecting all calls where the static analysis found a link between user input and `extract`. The source code surrounding the `extract` call was read and potential vulnerabilities stemming from overwriting variables noted. Thus, the list cannot be reproduced or verified automatically in this artifact.

The dataset of PHP applications from GitHub was created by Wessels *et al.* for their paper on "SSRF vs. Developers: A Study of SSRF-Defenses in PHP Applications" (USENIX Security '24), which produced the PHP-CPG framework our analysis is based on. We were granted access to the dataset, but since it was not created for this paper, it is not in scope to be released with this work.

The artifact contains a summary of the number of vulnerable projects and `extract` calls at the top. Following is a list of vulnerable `extract` calls in every project with markings of what type of vulnerability is possible when abusing the user input passed to `extract`, indicated by a “1” in the corresponding column. The sums of each column are displayed above the column. One project can use `extract` multiple times, which leads to the same project appearing multiple times. On the right of each row a link to the line with the `extract` call in the file on GitHub is provided and a description of target variables to overwrite to trigger the vulnerabilities.