Artifact abstract for the paper "Not In My Git Yard: Catching Backdoors at Commit and Release Time"

Dimitri Kokkonis
dimitri.kokkonis@cea.fr
Université Paris-Saclay
CEA, List
Palaiseau, France

Michaël Marcozzi
michael.marcozzi@cea.fr
Université Paris-Saclay
CEA, List
Palaiseau, France

Stefano Zacchiroli
stefano.zacchiroli@telecom-paris.fr
LTCI
Télécom Paris
Institut Polytechnique de Paris
Palaiseau, France

ACM Reference Format:
Dimitri Kokkonis, Michaël Marcozzi, and Stefano Zacchiroli. 2026. Artifact abstract for the paper "Not In My Git Yard: Catching Backdoors at Commit and Release Time". In *Proceedings of 41st IEEE/ACM International Conference on Automated Software Engineering (ASE 2026).* ACM, New York, NY, USA, 1 page. https://doi.org/XXXXXXX.XXXXXXX

1 PAPER TITLE

This artifact accompanies the paper *Not In My Git Tree: Catching Backdoors at Commit and Release Time.*

2 LINK TO THE ACCEPTED PAPER

A copy of the accepted version of the paper is available at https://zenodo.org/records/19337349 (`preprint.pdf`).

3 PURPOSE

This artifact enables evaluating Lily, a new approach and tool which allows for early detection of backdoor injections in the CI and release vetting pipelines, over a subset of the Rosarum benchmark [1], with a total of 545 unique version pairs over 13 software projects. More precisely, this artifact makes it possible to reproduce the evaluation results reported in our ASE'26 paper about Lily, as they are detailed in tables 3–6 of this paper.

4 BADGE

We are applying for the **Available**, (**Functional**) and **Reusable** badges, for the following reasons:

Available: the artifact satisfies the requirements of this badge, as it has been fully archived on the Zenodo public archival repository with an associated unique identifier: https://zenodo.org/records/19337349 (DOI: https://doi.org/10.5281/zenodo.19337349).

(Functional &) Reusable: the artifact satisfies the requirements of these badges, as it is:

Documented (carefully): the artifact and underlying code are thoroughly documented, covering not only the reproduction of the results of the paper, but also complete guides to reusing or extending the Lily and the experiment infrastructure of the artifact for other uses or purposes.

Consistent: the artifact allows to recompute the data discussed in the evaluation section of the paper from scratch and pretty-print them as they appear in the paper. It offers both a complete and partial reproduction option, to allow for partial verification of the paper's results in a reasonable amount of time.

Complete: all of the components developed to produce the paper's results are provided. This includes the full Lily toolchain (along with its development history and documentation), with full instructions on how to use it.

Exercisable: the artifact includes documented Bash and Python scripts that can either completely or partially reproduce the tables and diagrams of the paper, such that a reviewer may directly compare them with the tables and diagrams shown in the paper. While these scripts can be easily adapted to other uses or purposes, we also provide detailed instructions for reusing and repurposing both Lily and the experiment infrastructure of the artifact.

5 TECHNOLOGY

Evaluating the artifact requires the following:

Skills: basic familiarity with Docker and Unix command-line tools.

Hardware: an x86_64 machine with at least 8 cores, 32 GiB of RAM and 50 GiB of free disk space (for partial reproduction) or 170 GiB of free disk space (for complete reproduction).

Software: an installation of Docker (*Desktop* or *Engine*).

6 PROVENANCE

The artifact can be obtained at https://zenodo.org/records/19337349 (version `0.2.5`).

7 INSTRUCTIONS

Detailed instructions are provided at https://zenodo.org/records/19337349 (`README.md`). It is recommended that the reviewers follow the *Guided partial reproduction* section, but additional sections are provided for evaluating a selected subset (see *Unguided partial reproduction*) or the full set of experiments of the paper (see *Complete reproduction*).

REFERENCES

[1] Dimitri Kokkonis, Michaël Marcozzi, Emilien Decoux, and Stefano Zacchiroli. ROSA: Finding Backdoors with Fuzzing. In *2025 IEEE/ACM 47th International Conference on Software Engineering (ICSE)*, pages 2816–2828, 2025.

ASE 2026, 12-16 October 2026, Munich, Germany

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/XXXXXXX.XXXXXXX